3-27-2013 12:00 AM

Fuzzy Expert-COCOMO Risk Assessment and Effort Contingency Model in Software Project Management

Ekananta Manalif
The University of Western Ontario

Supervisor
Luiz Fernando Capretz, PhD, P.Eng
The University of Western Ontario

Graduate Program in Electrical and Computer Engineering
A thesis submitted in partial fulfillment of the requirements for the degree in Master of Engineering Science

FUZZY EXPERT-COCOMO RISK ASSESSMENT AND EFFORT CONTINGENCY MODEL IN SOFTWARE PROJECT MANAGEMENT

(Thesis format: Monograph)

by

Ekananta Manalif

Graduate Program in Software Engineering
Department of Electrical and Computer Engineering

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Engineering Science

The School of Graduate and Postdoctoral Studies
The University of Western Ontario
London, Ontario, Canada

Abstract

Software development can be considered to be the most uncertain project when compared to other types of projects due to the uncertainty in the customer requirements, the complexity of the process, and the intangible nature of the product. In order to increase the likelihood of success in managing a software project, the project manager(s) must invest more time and effort in the project planning phase. The two main activities in project planning phase are effort estimation and risk management which have to be executed together because the accuracy of the effort estimation is highly dependent on the size and nature of the project risks.

However, as a common practice in a software development project, effort estimation and risk management are often disconnected from each other and most of the software effort estimation methodologies, which include the COCOMO model, provide a fixed value rather than an approximation value (base value and allowance value), and consequently the existing effort estimation approach has failed to provide a reliable reference for project manager due to its lack of accuracy.

This thesis introduces the Fuzzy Expert-COCOMO Model, the Risk Assessment and Effort Contingency Model based on COCOMO cost factors and fuzzy technique, which has the capability to not only integrate the effort estimation and risk assessment activities into the initial project planning phase but also to provide the essential information about the estimated effort, the project risks, and the effort contingency allowance required to accommodate the identified risk.

A validation of this model using project data set shows that the new approach is capable of improving the sensitivity of project risk assessment using Expert-COCOMO methodology and also provides a higher level of effort prediction performance compare to the existing COCOMO-II effort estimation.

Keywords

Effort Estimation, Risk Assessment, Effort Contingency Allowance.

Dedicated to

My dearest wife, Eva

My sons, Farrell, and Nito

They have been the biggest motivator and inspiration in my life.

Acknowledgement

First of all, I would like to offer my sincerest gratitude to my supervisors, Dr. Luiz Fernando Capretz and Mr. Danny Ho, for giving me the opportunity to carry out my graduate study at University of Western Ontario and the opportunity to work under their supervision. Their encouragement, guidance and support throughout all stages of the thesis process enabled me to conduct the research and finalized this thesis. I am indebted to them more than they realize, without their support, I would not be possible to complete this program.

I also gratefully thank to my colleagues and friends, especially Dr. Ali Bou Nassif who provided me advice related to my research content. He kindly granted me his time to answer some of my questions. I have benefited from all his advice and information.

Moreover, I would like to show my appreciation to my wife and my children, whose love and reassurance allowed me to return to school and encouraged me through all of the difficult path in my academic journey. My mother and families, as I greatly appreciate their moral support and their prayers for my success.

Then, I should acknowledge all faculties who were part of my education program, Dr. Jagath Samarabandu, Dr. Miriam Capretz, and Dr. Nazim Madhavji. I also personally appreciate our graduate assistant Chris Marriott very much who easily lent me a hand when I needed.

Lastly, I offer my regards to everyone who has been a part of this journey and has supported me in any respect during the completion of my study.

Thank you for all of you...!

Table of Contents

List of Tables

List of Figures

List of Appendices

Chapter 1

Introduction

1.1 Motivation

Software projects development can be considered to be the most uncertain and complex project when compared to other types of engineering projects, because their activities involve an intangible product and these are continually changing in response to customers' requirements and the development of new technology. The 2009 Standish Group Chaos report [1] showed that only 32% of such projects succeeded. That is, they were delivered on time, within budget, and had the required features and functions: 44% did not meet these three requirements, and 24% failed, i.e., they were cancelled prior to completion or were delivered and never used. The results of a study conducted by TATA Consultancy Services in 2007 to 800 senior IT managers from the UK, the US, France, Germany, India, Japan, and Singapore were similar to those of the Standish Group report: 62% of projects failed to meet their schedule, 49% experienced budget overruns, 47% experienced higher maintenance costs, and 41% failed to deliver the expected Return on Investment (ROI) [2].

Based on the results of several investigations of software development projects, the main areas responsible for project failure were found to be as follows: project goal setting, project scheduling, project staffing (availability and capabilities), customer requirements, unmanaged risks, improper project execution, stakeholder politics, and commercial pressures [3][4].

Project managers usually use the project management techniques, which are defined as "the application of knowledge, skills, tools and techniques to project activities in order to meet project requirements" and which involve the following five inter-related development processes: initiating, planning, executing, monitoring and controlling, and closing [5]. Based on the above list of the most influential causes of software project failure, the success of a software project will be highly dependent on the Project Planning

Phase, which involves those activities that determine a project's scope, scheduling, cost, resources, and risks. Therefore, while this phase is critical in all cases of project management [6], it is especially so in a software development project.

The main activities in the software project planning phase are effort estimation and risk management [7]. These two activities together with quality estimation become the major issues in the success of software development project and the accuracy of the results will provide the great support in project execution phase [8]. Software effort estimation calculates the effort necessary to complete the project, in term of scheduling, acquiring resources, and meeting costs. The cost element in a software development project depends on the making of several cost estimates, while risk management activities include identifying, addressing, and eliminating software project risks before undesirable outcomes occur.

Software effort estimation is an essential activity in the planning phase due to its role in helping project managers with respect to the budgeting, scheduling, and allocation of resources. Risk management also plays a vital role here, considering the fact that a software project will be used in an environment where the results are intangible and subject to a higher level of uncertainty compared to the other types of projects.

In the Project Planning Phase, risk management activities focus mostly on risk assessment, which is a discovery process of identifying the potential risks, analyzing or evaluating their risk effects, and prioritizing the risks. Risk identification activity focuses on enumerating the possible risks, creating a risk statement, and establishing the context of the possible risks as deliverables. Based on the risk statement and the risk context, all aspects of the risks can then be analysed and prioritized, so that a project manager can determine where action should be taken to manage such risks [9]. Hence, the risk assessment phase will provide the information about the number of risks and an estimate of the risk-exposure relationship of each.

In order to support project managers in a software development project, several models have been developed to assist in the Effort Estimation and Software Risk Assessment. The most significant effort estimation models that have been used in software development projects are the Constructive Cost Model (COCOMO) [10], the System

Evaluation and Estimation of Resource Software Evaluation Model (SEER-SEM) [11], and the Software Life Cycle Management (SLIM) model [12]. The COCOMO model, which was developed by Barry Boehm in the 1980s, is the most widely used estimation model for software project. Several other methods have also been developed to support risk management in software project development. These include: Boehm Risk Management [13], Jone's Risk Management [14], the Software Engineering Risk Model (SERIM) [15], the Software Risk Analysis and Management (SRAM) [16], Hall's P^2I^2 Formula [17], and the Software Risk Management Framework (SRM) from Software Engineering Institute (SEI) [18].

As compared to an effort estimation activity, risk management, especially risk assessment in software project planning, is rarely found and is often difficult to implement because of the scarcity of experts, the unique project characteristics, the lack of sufficient time to do a thorough analysis, and perceived as effort intensive and costly [19][20].

Cost is the single most important factor in managing software projects and project risks have an adverse impact on the estimated cost of a software project. Inaccuracy in effort estimation will be costly for the development team and may result in loss of business [21]. To address this problem, some attempts have already been made to integrate cost estimation with risk management [22][23][24]. But still, these approaches have failed to explain the effect of identified risk on the estimated effort.

Applying the Expert-COCOMO model is one efficient approach to software project risk management [19], because it leverages the existing knowledge and expertise from previous effort estimation activities based on COCOMO to assess the level of risk in a new software development project. However, Expert-COCOMO Model has a limitation because it cannot effectively deal with imprecise and uncertain information in the form of linguistic terms such as: Very Low (VL), Low (L), Nominal (N), High (H), Very High (VH) and Extra High (XH). On the other hand, because risk can be considered to be an abstract and fuzzy concept [24], users are having difficulties defining risk more accurately for deeper analysis. This limitation might cause problems with the application of model and might affect the risk assessment accuracy using the Expert-COCOMO approach.

Recent developments in soft computing have provided the software engineering community with promising techniques such as Fuzzy Logic, which aims to serve as a tool for dealing with uncertainty, imprecision, and complex problems that are difficult to solve quantitatively. A fuzzy Inference model is a powerful tool for solving problems that involve imprecise and uncertain information, such as effort estimation and risk assessment, and provides a solution that is easy to understand and interpret.

In this thesis, we propose using a Fuzzy-ExCOM (Fuzzy Expert-COCOMO) Model, which is a novel risk assessment model that combines the advantages of a fuzzy system with Expert-COCOMO methodology for risk assessments in a software project. Validation of this approach in industry project data shows that the proposed model provides a higher level of sensitivity in risk identification compared to the original methodology. Another feature of our model is that it allows a project manager to estimate the contingency level of effort estimation based on identified project risk and software size. The information generated from this model can then be used as a decision support system for an individual project manager in conducting deeper risk assessment and developing risk mitigation approaches.

1.2 Research Questions

The research presented in this thesis is to find answers to the following research questions due to the limitations on the existing approach, such as:

RQ-1: How the fuzzy technique improves the sensitivity of software project risk assessment using Expert-COCOMO methodology?

RQ-2: How the identified software project risks affected the COCOMO effort estimation approach?

RQ-3: How the identified software project risks can be used to improve the COCOMO effort estimation approach?

1.3 Research Objectives

This research attempts to improve the risk assessment and effort estimation accuracy by:

- Applying a fuzzy inference system to an Expert-COCOMO model to improve the sensitivity in risk assessment.
- Developing and validating a new “Fuzzy-ExCOM Model”, to estimate the effort estimation contingency based on software project risk and software size.

1.4 Thesis Contributions

This research makes two main contributions. First, it creates a new software risk assessment method based on a combination of a fuzzy inference system and Expert-COCOMO for early project risk identification, risk prioritization, and the creation of a risk mitigation plan based on project cost factors.

Second, it introduces the “Fuzzy-ExCOM Model” to be used to increase the sensitivity of software risk assessments and effort estimations. The proposed model provides a list of software project risks and effort estimation contingency based on project cost factors, project risks, and software size.

1.5 Thesis Organization

This thesis is structured as follows:

Chapter 2 (Literature Review) discusses the background of software project management especially in the project planning stage, which focuses more on software project effort estimation and project risk management. The discussion covers topics such as project management, software risk management, software estimation, and soft-computing tools.

Chapter 3 (Fuzzy-Expert-COCOMO Model) introduces the Fuzzy Expert-COCOMO (Fuzzy-ExCOM) Model as an improvement over Expert-COCOMO for software project risk assessment. In this chapter, a discussion about the applicable Effort Estimation Accuracy Level using the Fuzzy-ExCOM Model is presented.

Chapter 4 (Evaluation by Project Data) presents the evaluation design results and implementation of the Fuzzy-ExCOM Model using COCOMO public project data set and industrial project data set.

Chapter 5 (Conclusions and Future Work) summarizes the conclusions reached based on the research activity and describes the direction of future work in the area of software risk assessment.

Chapter 2
Literature Review

This chapter presents background information related to the problems identified in this research which related to the Effort Estimation and Risk Assessment in software project planning phase. The main topics discussed are related to the description, advantage, and disadvantage of software project management, effort estimation, risk management, and the fuzzy logic technique as the main tool that was used in this research.

This chapter is organized as follows: Section 2.1 provides an overview of project management; the effort estimation is described in section 2.2, while section 2.3 describes the risk management. The Expert-COCOMO risk assessment method is discussed in section 2.4, section 2.5 describes the Fuzzy Logic technique, and the chapter summary is described in section 2.6.

2.1 Project Management

Project Management can be considered as a new approach in the domain of management knowledge and has rapidly gained popularity since the first certification in project management was launched by the Project Management Institute (PMI) in 1984. A good indicator of this growth is the rapid expansion of PMI membership as a professional organization for project managers, which has grown from 93,000 members in 2002 to more than 600,000 members in 2012 [25]. The importance of project management is also supported by the fact that a project, which is defined as "a temporary endeavor undertaken to create a unique product, service, or result" [5], involves the most important activities for the implementation of an organizational strategy that is designed to achieve specific objectives in modern organizations [26].

PMI has also defined project management as "the application of knowledge, skills, tools and techniques to project activities in order to meet project requirements" [5]. The project management goal is to ensure the creation of a high quality end product (scope) within a specific time (schedule) and within a set budget by balancing the demands of the various project management variables such as people, scope, tools and techniques. The relationship of scope, schedule, and budget in project management, as illustrated in Figure 2.1, is sometimes referred to as the triple constraint of project management.

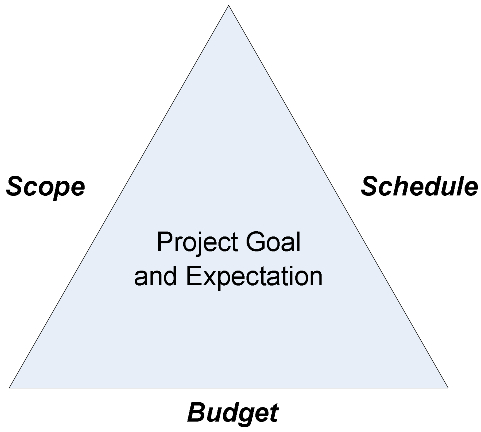


Figure 2.1 Triple Constraint of Project Management [27]

2.1.1 Project Management Body of Knowledge (PMBOK®)

A Guide to the Project Management Body of Knowledge (PMBOK® Guide), is PMI's standards document, defines nine knowledge areas with sub-activities that help the project manager with the simultaneous visibility to oversee and manage a project successfully. The nine project management knowledge areas are [5]:

1. Project Integration Management
2. Project Scope Management
3. Project Time Management
4. Project Cost Management

5. Project Quality Management
6. Project Human Resources Management
7. Project Communication Management
8. Project Risk Management
9. Project Procurement Management

The above project management knowledge areas have 42 corresponding sub-activities that are logically grouped into the following five processes:

1. Initiating Process Group
2. Planning Process Group
3. Executing Process Group
4. Monitoring and Control Process Group
5. Closing Process Group

A mapping of each Project Management Process Group with its corresponding Project Management Knowledge Area is provided in Table 2.1.

2.1.2 Software Project Management

Since the introduction of the term, software crisis, at the 1968 NATO Software Engineering Conference, the software community has applied a systematic, disciplined, and quantifiable approach to the development, operation, and maintenance of a standardized software system [28]. This approach is known as Software Engineering which focuses on studying and investigating the application of engineering approaches to software development and also defined as a systematic approach to the analysis, design, assessment, implementation, testing, maintenance, and re-engineering of software systems [29].

Table 2.1 Project Management Process Group and Knowledge Area Mapping [5]

Knowledge Area	Project Management Process Group				
	Initiating Process Group	**Planning Process Group**	**Executing Process Group**	**Monitoring & Controlling Process Group**	**Closing Process Group**
Project Integration Management	• Develop Project Charter	• Develop Project Management Plan	• Direct and Manage Project Execution	• Monitor and Control Project Work • Perform Integrated Change Control	• Close Project or Phase
Project Scope Management		• Collect Requirements • Define Scope • Create WBS		• Verify Scope • Control Scope	
Project Time Management		• Define Activities • Sequence Activities • Estimate Activities Resources • Estimate Activity Durations • Develop Schedule		• Control Schedule	
Project Cost Management		• Estimate Costs • Determine Budget		• Control Costs	
Project Quality Management		• Plan Quality	• Perform Quality Assurance	• Perform Quality Control	
Project Human Resources Management		• Develop Human Resources Plan	• Acquire Project Team • Develop Project Team • Manage Project Team		
Project Communication Management	• Identify Stakeholder	• Plan Communication	• Distribute Information • Manage Stakeholder Expectations	• Report Performance	
Project Risk Management		• Plan Risk Management • Identify Risks • Perform Qualitative Risk Analysis • Perform Quantitative Risk Analysis • Plan Risk Responses		• Monitor and Control Risks	
Project Procurement Management		• Plan Procurements	• Conduct Procurements	• Administer Procurements	• Close Procurements

Software Project Management is the art and science of planning and directing software projects [30]. It is the software engineering study area that governs the implementation of Project Management principles for a planned, monitored, and controlled software project. Software Project Management involves: configuration, resource management, and risk management as well as the development and implementation of a resource allocation plan, a software development plan, and a software development strategy.

In line with the increasing importance of a software system for a modern organization to service its customer's needs, Software Project Management becomes even more important in managing current software projects which consists of millions of lines of code (LOC) within the uncertainty environment such as: the uncertainty of the input information (customer requirements), the complexity of the process, and the intangible nature of the product.

2.1.3 Software Project Failure

Based on the project management definition, project failure can be understood as being the failure to meet the project goals, which relate to a project's scope, time constraints (schedule), and budget. As the most risky type of project compared to other projects, a software project can be considered to be the most difficult type to manage and subsequently achieve project goals.

Based on the results of the investigations of several software development projects, the main areas responsible for project failure and their relationship to the various project management process groups can be described as follows [3][4]:

- Project Planning Process: *project goal setting; project scheduling (unrealistic schedule); inappropriate project staffing (availability and capabilities); badly defined customer requirements and changing requirements during development; inaccurate estimates of required resources; unmanaged risks; use of immature technology.*

- Project Execution Process: *improper project execution; inability to handle the project's complexity; inefficient development practices; inferior quality work.*
- Project Monitoring and Controlling Process: *lack of good communication links among customers, developers, and users; inadequate reporting of the project's status.*

Of the three project process groups (planning, execution, and monitoring/controlling), the project planning process group is one of the most critical elements in managing a project because it involves determining a project’s scope, scheduling, cost, available resources, and project risks [6].

In order to increase the chance of success in managing a software project that meets the established requirements of scope, time and budget, the project manager(s) must invest more effort in the project planning phase, which involves such primary elements as effort estimation and risk management [7].

2.2 Effort Estimation

Software effort estimation determines the amount of effort necessary to complete a software project, in terms of its scheduling, the acquiring of resources, and the meeting of budget requirements. This is an essential activity in the software project planning phase because major problems usually surface in the first three months of a software development project and are related to the hasty scheduling, irrational commitments, and unprofessional estimating techniques [31].

In the early stages of a software development life cycle, effort estimation plays a critical role in helping project managers identify the demands in a software development project with respect to the budgeting, scheduling, and the allocation of resources. Several techniques have been used to estimate effort in software development project and can be classified in two main categories, the pragmatic approach and the modeling approach [32].

2.2.1 Pragmatic Approach

In this category, the estimation approaches are not based on theoretical method or formal method and tend to estimate software project effort base on the individual expertise and historical data. Some approaches in this category are: rule of thumb, analogy, and expert judgement. The approaches in this category are usually very subjective, lack of standardization, and cannot be reusable [33].

2.2.2 Modeling Approach

There are two categories of effort estimation based on modeling approaches; they are effort estimation based on Algorithmic Models and effort estimation models based on Machine Learning [34].

The Algorithmic Models can be considered as the most popular approach in software project effort estimation. Several models that have been used in software development projects are COCOMO [10], SEER-SEM [11], and SLIM [12].

SEER-SEM, which was developed by Galorath and Evans in 1990, is a powerful and sophisticated model, which includes a variety of tools for several kinds of estimation activities in software development projects as well as effort estimation [35]. This model accommodates several knowledge bases (KBs) as inputs to set baselines for over 50 parameters that will impact the estimation output. The Knowledge bases in SEER-SEM are related to platform, application, acquisition method, development method, development standards, and class, which can be selected based on the project requirements.

SLIM is an empirical software effort estimation model developed by Lawrence Putnam in the 1970s. The model was developed based on a collection of software project data (size and effort); it calculates the associated effort using an equation that fits the original data. The SLIM model can be considered to be one of the earliest and well known software effort estimation models in the software industry.

COCOMO, which was developed by Barry Boehm in the 1980s, is the most popular and most widely used estimation model for software projects. COCOMO estimates the software project effort based on the scale and cost factors of a software project.

The Machine Learning Models in effort estimation have been recently used to complement or improve the algorithmic model approaches. Several works related to machine learning approach are: the utilization of neuro-fuzzy technique for SEER-SEM [35], application of neuro-fuzzy technique for COCOMO [36], and implementation of artificial neural network for effort estimation [37].

2.2.3 COCOMO Model

The COCOMO model is based on source line of code (SLOC), which is used to estimate the cost, effort requirements, and scheduling of a software development project. The first version of the COCOMO model is known as COCOMO'81 and has three levels that reflect the degree of detail of the estimation analysis. The first level (basic) provides an initial, rough estimate; the second level modifies the equation using a number of project and process multipliers; the third level is the most detailed and produces specific estimates for different phases of a project.

The COCOMO model has been continuously evolving to address the rapid changes in software technology and the software development process. In the 1990s, COCOMO II along with three sub models was introduced as an evolution from COCOMO'81 and accommodates the new approaches to software development. The three models in COCOMO II are [38]:

- *The Applications Composition Model*: Used in the early stage of software development and estimates the effort required in software development using size estimates based on object points and a simple size/productivity formula.

- *The Early Design Model*: Used after project requirements and basic architecture have been stabilized. The formula follows the standard form with a simple set of multipliers associated with it.

- *The Post Architecture Model*: Used during the software construction and the effort estimation stage. This model uses a more extensive set of multipliers reflecting personnel capability as well as product and project characteristics.

The Post Architecture Model is the most detailed of the COCOMO-II models and is expressed in the following formula:

$$\text{Effort} = \boldsymbol{A} \times (Size)^{\boldsymbol{B}+\boldsymbol{C}} \times \prod_{i=1}^{17} \mathbf{EM}_i \qquad \dots \ (2.1)$$

$$\mathbf{C} = 0.01 \times \sum_{i=1}^{5} \boldsymbol{SF}i$$

Where **Effort** is the estimated software development effort in staff-months; ***A*** = *2.94* and ***B*** = *0.91* are the baseline calibration constants; ***Size*** is the size of the software project measured in terms of KSLOC (thousands of Source Lines of Code); ***SF****i*'s are the five Scale Factors; and ***EM****i*'s are the seventeen Effort Multipliers.

The five scale factors and seventeen effort multipliers of the COCOMO-II Post Architecture model are called cost drivers, which are defined in Table 2.2. The twenty-two cost drivers are measured qualitatively by selecting a rating from the following well-defined rating levels, i.e. Very Low (VL), Low (L), Nominal (N), High (H), Very High (VH) and Extra High (XH). Each cost driver has four to six rating levels and each rating level of every cost driver is associated with a number that is identified with a parameter value, which is used in the COCOMO formula. The parameter values are shown in Table 2.3 for the COCOMO-II version calibration.

Table.2.2. COCOMO-II Post Architecture Cost Drivers [42]

Scale Factor	Symbol	Explanation
PREC	SF1	**Precedentedness**. Reflects the previous experience of the organization with this type of project. Very Low means no previous experience; Extra high means that the organisation is completely familiar with this application domain.
FLEX	SF2	**Development Flexibility**. Reflects the degeree of the flexibility in the development process. Very low means a prescribed process is used; Extra high means that the client sets only general goals.
RESL	SF3	**Architecture/Risk Resolution**. Reflects the extent of risk analysis carried out. Very low means little analysis; Extra high means a complete ad thorough risk analysis.
TEAM	SF4	**Team Cohesion.** Reflects how well the development team members know each other and work together. Very low means very difficult interactions; Extra high means an integrated and effective team with no communication problems.
PMAT	SF5	**Process Maturity.** Reflects the process maturity of the organization. The computation of this value depends on the CMM Maturity Questionnaire, but an estimate can be achieved by subtracting the CMM process maturity level from 5.

Effort Multipliers	Symbol	Description
		Product Factors
RELY	EM1	Required Software reliability
DATA	EM2	Size of Database used
CPLX	EM3	Complexity of system modules
RUSE	EM4	Required Reusability
DOCU	EM5	Extent of Documentation required
		Platform Factors
TIME	EM6	Excution-time constraints
STOR	EM7	Main storage constraints
PVOL	EM8	Volatility of development platform
		Personnel Factors
ACAP	EM9	Capbility of project analyst
PCAP	EM10	Programmer Capability
PCON	EM11	Personnel Continuity
APEX	EM12	Application Experience
PLEX	EM13	Platform Experience
LTEX	EM14	Languange and Tool Experience
		Project Factors
TOOL	EM15	Use of software tools
SITE	EM16	Extent of mulitsite working and quality of inter-site communications
SCED	EM17	Development schedule compression

Table.2.3. COCOMO-II Scale Factors and Efforts Multipliers [42]

Drivers		Symbol	Scale Factors					
			Very Low	Low	Nominal	High	Very High	Extra High
Precedentedness	**PREC**	SF1	6.20	4.96	3.72	2.48	1.24	0.00
Development Flexibility	**FLEX**	SF2	5.07	4.05	3.04	2.03	1.01	0.00
Architecture/Risk Resolution	**RESL**	SF3	7.07	5.65	4.24	2.83	1.41	0.00
Team Cohesion	**TEAM**	SF4	5.48	4.38	3.29	2.19	1.10	0.00
Process Maturity	**PMAT**	SF5	7.80	6.24	4.68	3.12	1.56	0.00

		Symbol	Effort Multiplier					
			Very Low	Low	Nominal	High	Very High	Extra High
Product Factors								
Reliability required	**RELY**	EM1	0.82	0.92	1.00	1.10	1.26	
Database size	**DATA**	EM2		0.90	1.00	1.14	1.28	
Product Complexity	**CPLX**	EM3	0.73	0.87	1.00	1.17	1.34	1.74
Required Reusability	**RUSE**	EM4		0.95	1.00	1.07	1.15	1.24
Documentation Needs	**DOCU**	EM5	0.81	0.91	1.00	1.11	1.23	
Platform Factors								
Excution-time constraints	**TIME**	EM6			1.00	1.11	1.29	1.63
Main storage constraints	**STOR**	EM7			1.00	1.05	1.17	1.46
Platform volatility	**PVOL**	EM8		0.87	1.00	1.15	1.30	
Personnel Factors								
Analyst Capability	**ACAP**	EM9	1.42	1.22	1.00	0.85	0.71	
Programmer Capability	**PCAP**	EM10	1.34	1.16	1.00	0.88	0.76	
Personnel Continuity	**PCON**	EM11	1.29	1.10	1.00	0.90	0.81	
Application Experience	**APEX**	EM12	1.22	1.10	1.00	0.88	0.81	
Platform Experience	**PLEX**	EM13	1.19	1.12	1.00	0.91	0.85	
Languange and Tool Experience	**LTEX**	EM14	1.20	1.10	1.00	0.91	0.84	
Project Factors								
Use of software tools	**TOOL**	EM15	1.17	1.09	1.00	0.90	0.78	
Multi-site development	**SITE**	EM16	1.22	1.09	1.00	0.93	0.86	0.8
Required Development Schedule	**SCED**	EM17	1.43	1.14	1.00	1.00	1.00	

2.2.4 Estimation Model Limitation

Estimation activity is defined as the activity for a rough calculation about value, number, or something [39]. Result from estimation process is approximation value and the activity typically means finding upper or lower bounds of a quantity that cannot readily be computed precisely. Estimation is very useful especially if it involves the incomplete or uncertain parameters.

Since an estimation value represents an approximation value, every estimation result must have a contingency [40]. The ideal estimation calculation provides base-value with contingency allowance which covers the risks and assumption for certain estimation calculation. The three core purposes served by contingency allowance in the project plan are: accounting errors and omissions, accommodating scope change and modification, and unknown conditions anticipation [41].

However, every software effort estimation methodologies including COCOMO do not provide contingency allowance for their estimation value. Most software effort estimation methodology provides the fix estimation value instead of approximation value. In the software project planning practice, the fix effort estimation value is not very useful because project manager do not have a proper guidance to set the project budget.

Since the contingency allowance is highly related to software risk, the ability in providing the effort contingency allowance value is highly dependent on the ability to conduct the software risk assessment as the integral part of effort estimation. The common practice to set the contingency allowance in software development project is based on subjective judgment and experience of the project manager and the activity is totally separate with estimation activity and consequently the impact of an identified risk on the accuracy of effort estimation is difficult to identify.

2.3 Risk Management

Risk always brings uncertainty and is inherent in every project that has the potential for substantial loss. Two intrinsic properties of risk are: uncertainty and loss [43]. Uncertainty relates to something with which we are not totally familiar and loss can be understood as an unfavourable outcome or lost opportunity. In project management, a risk can be understood as a possible event that would have a negative impact on the outcome of a project if it were to occur.

Risk management provides a clear and structured approach to identifying and managing the risks to which a project is exposed, which is based on the creation and implementation of an effective plan to either prevent losses or to reduce their impact if they should occur. Effective risk management practice does not eliminate risk. It merely reduces risk by providing project managers with a means by which they can measure and prioritize inherent project risks in order to make well informed decisions with respect to identifying appropriate actions.

2.3.1 Risk Definition and Standard

Risk management is a rapidly developing discipline that is widely applied in the areas of finance, engineering, security, various industrial processes, actuarial practice, public health and safety, and project management. As a new discipline, risk management has a wide variety of perspectives and descriptions that have their origins in such organizations as: The International Organization for Standardization (ISO), The Institute of Risk Management (IRM), The Institute of Electrical and Electronics Engineers (IEEE), various actuarial societies, and The Project Management Institute (PMI).

In general, the risk management standard refers to the terminology that is set out by the ISO that is documented in the “ISO/IEC Guide-73 Risk Management – Vocabulary” that is part of the family of ISO-31000 standard documentation [44]. In 2009, IRM, the Association of Insurance and Risk Management (AIRMIC) and ALARM (The National Forum for Risk Management in the Public Sector) published a new document called “A Structured Approach to Enterprise Risk Management (ERM) and the Requirements of ISO-31000” as a guideline for risk management implementation in various business levels [45] following the publication of ISO-31000.

The ISO-31000 defines and describes the scope, principles, framework, and process of risk management that can be applied at all levels of an organization. The relationship between the principles, framework, and process involved in risk management is shown in Figure 2.2. ISO-31000 defines risk as “the effect of uncertainty on objectives, whether positive or negative”, and risk management as the “coordinated activities to direct and control an organization with regard to risk” [46].

At the project management level, the risk management standard refers to the standard defined in the PMBOK® which defines risk as “an uncertain event or condition that, if it occurs, has [either a] positive or negative effect on the project objectives” [47]. The Project Risk Management knowledge area is one of nine knowledge areas addressed in PMBOK® that describes the process of risk management planning, identification, analysis, response, monitoring, and control that acts as a guide for implementing project risk management.

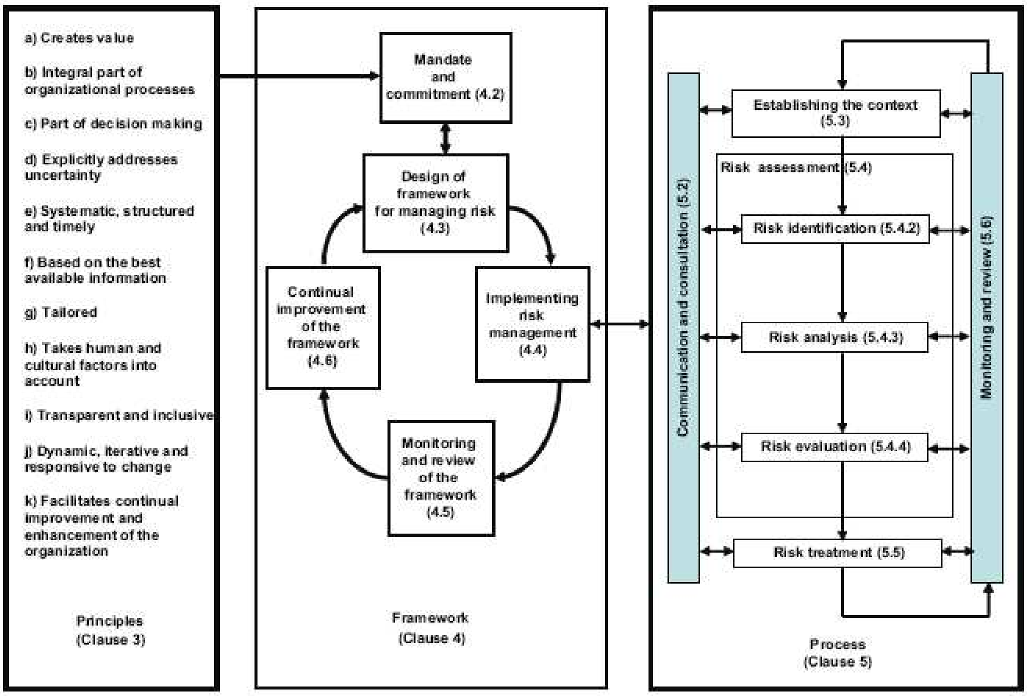


Figure 2.2 Risk Management Principles, Framework, and Process Relationship [46]

In addition to the PMI standard for project risk management, the guide for risk management in a software project also refers to the IEEE-Std-1540 and the Software Engineering Institute (SEI) Technical Report-1996 (CMU/SEI-96-TR-012).

The IEEE-Std-1540 prescribes a continuous risk management process to be followed in a software project, which can be applied throughout the software life cycle. This standard defines risk as “the likelihood of an event, hazard, threat, or situation occurring and its undesirable consequences; a potential problem” [48]. This standard not only applies to specific aspect of the software life cycle process model but also supports the whole process, starting from acquisition, supply, development, operation, and maintenance of software products and services.

The SEI Technical Report 1996 presents a holistic vision of the risk-based methodologies in Software Risk Management (SRM). In conjunction with the SRM framework, SEI defines software risk as “a measure of the probability and severity of adverse effects inherent in the development of software that does not meet its intended functions and performance requirements” [49].

Based on several definitions provided above, software risk can be understood as: *an uncertain event or condition that, if it occurs, has either positive or negative consequences with respect to the project objective to develop software with specific functions and performance requirements. Risk is inherent in software development and can be expressed as a combination of probability, or frequency of occurrence and severity.*

2.3.2 Project Risk Management

In the area of project management, risk management for almost any kind of project generally refers to the PMBOK®. The risk management objectives as set out by PMI are based on the need to increase the probability and impact of positive risk and decrease that of negative risk. This objective can be met through 5 processes, i.e., risk management planning, risk identification, risk analysis, risk response planning, and monitoring and control [47].

Risk management planning is the process followed to establish the means by which to conduct risk management activities and should be completed during the initial stage. Risk identification is the iterative process used to determine and document the risk(s) that may affect a project. The risk analysis, which is required in risk management, takes into consideration the fact that not all identified risk elements warrant a response. This process can be completed both qualitatively and quantitatively. The qualitative risk analysis is the process of prioritising risk for further actions based on a subjective evaluation of the probability and impact of each risk. If a qualitative risk analysis is based on subjective evaluation, then a quantitative risk analysis is the process of numerically analyzing the effect of identified risk with respect to the status of the overall project. The quantitative risk analysis result is usually used as a complement to the qualitative analysis. In project management, quantitative risk analysis is not the most important part and is not always required in risk management [50].

Risk response planning is the process following risk analysis that involves developing options and actions to enhance opportunities and to reduce threats from identified risks. Risk monitoring and control is the implementation stage of previous risk management processes and belongs to the Monitoring and Controlling Process Group. In this instance, the risk owner will respond to a risk trigger by implementing the appropriate contingency or fallback plan. The impact and benefit of risk management will be realized during this process because most of project problems are prevented at this stage [50].

In risk management, the Project Planning Process Group is the most critical element in determining the success of project risk management because four of the five risk management steps must be taken within this process group. The overview of Project Risk Management describes by PMI is shown in Figure 2.3 complete with the inputs, outputs, and required tools and techniques for each stage.

PMI’s risk management provides a suitable and complete guidance for project managers to manage risk in a generic project environment. However, this approach and guidance system is considered too general to be useful in a software design project [51]. This limitation is based on the fact that a software project can be considered to be the most uncertain, unique, and complex of projects when compared to other types of projects.

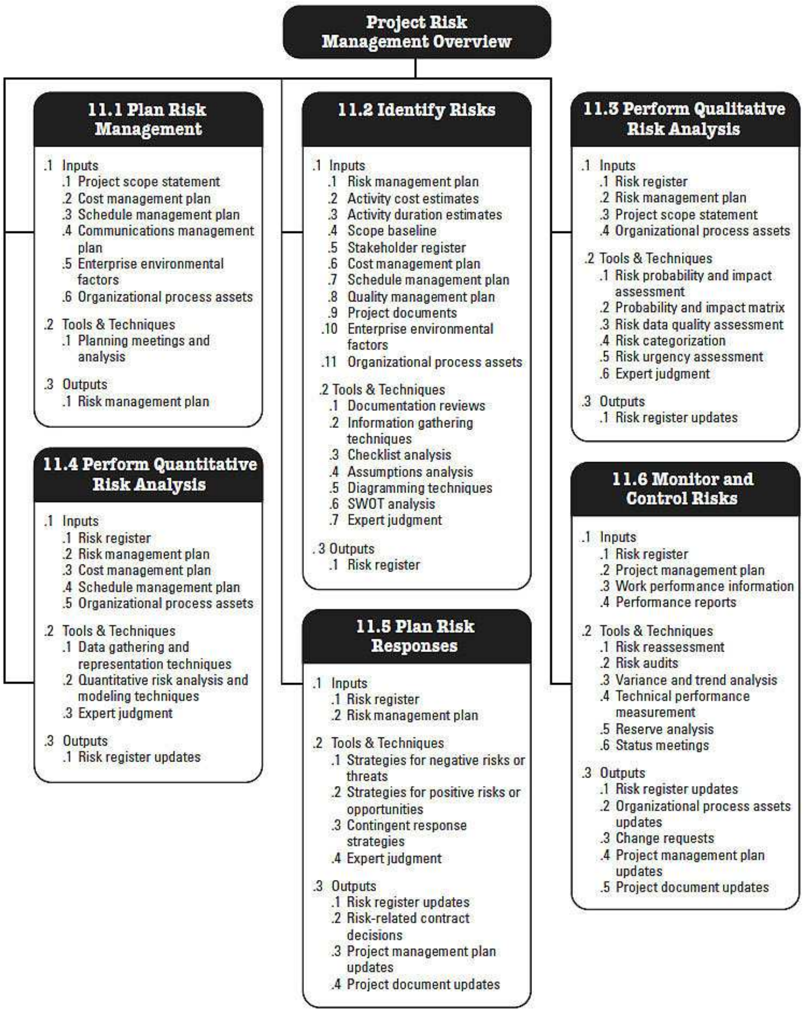


Figure 2.3 PMI's Project Risk Management [47]

2.3.3 Software Project Risk

The uncertain events, which create project risk, vary in every project according to the extent of project scope and technology. If the project scope is known and stable, and the technology to be used is familiar and proven, project uncertainty can be reduced by conducting effective planning sessions, which include the risk management planning stage [51].

The stability of scope and technology that is used in a software project is rarely found because each software project has unique challenges [52], which are:

a) Software is an abstract and complex product.

b) Both initial and final requirements are incomplete.

c) Technology changes rapidly and the related experience is never sufficiently comprehensive.

d) Best practices are not yet mature.

e) Software Development is an on-going research project.

f) Repetitive work is automated.

g) Construction is actually design.

h) Change is considered easy and inevitable.

The uniqueness and different characteristics of a software product as compared to other products create a high level of uncertainty in software project development, and consequently the software project itself becomes one of high risk.

In this situation, the generic approach to risk management should be either improved or modified so that it can be usefully applied to a specific software project. Several concepts have been introduced to support risk management activities in a software development project. These include such risk management approaches as those developed by Barry Boehm, Capers Jones, Dale Karolak, Robert Charette, Elaine Hall, and SEI.

2.3.3.1 Boehm's Risk Management

Boehm's risk management concept in software engineering has been used since the 1980s and was mentioned in the Software Engineering Economics handbook in 1981 [53]. He introduced the concept that risk mitigation in a software project can be achieved by identifying and resolving any software problems during the early software development life cycle (the requirements and design specification phase) [54].

In 1988, Boehm introduced the Spiral model for software development in an attempt to improve the waterfall model and reduce the likelihood of risk in the software development process. However, it was difficult to convince customers to use this model because this approach demanded a high level of risk assessment expertise and relied heavily on this expertise for the success of each project [55].

In 1991, the overall method of risk management was introduced [56]. This activity was divided into two phases, *Risk-Assessment* and *Risk-Control*.

Risk-Assessment is a discovery process of identifying the sources of risks, analyzing or evaluating the potential risk effects, and prioritizing these risks. The risk identification activity focuses on identifying the possible risks by use of checklists, decision driver analysis, assumption analysis, and decomposition. The list of the Top 10 Software Risk Items as shown in Table 2.4 is a common and popular tool that can be used in the risk identification activity. After the identification of individual risks, comes the item by item analysis to estimate the degree of exposure of each risk to the overall project. The final step in risk assessment is to prioritize the risks so that a project manager can determine where action should be taken to manage such risks. Risk prioritization can be done by assessing the risk probabilities from historical data and using Delphi or other group assessment techniques [57].

Risk-Control is a process of developing software risk resolution plans, monitoring the risk status, implementing a risk resolution plan, and resolving the risk issues by correcting potential deviations from the plan. The risk management planning activity will create a risk action plan, which will provide a basis for the risk resolution activity, and will describe the most likely scenarios and triggers for risk-tracking purposes. Risk resolution is the activity to implement or execute the risk management plan that was created based on such techniques as: prototyping, benchmarking, and simulation. Risk monitoring activities will track every risk based on the established plan or scenarios taken from the risk planning step and will provide an up-to-date risk status report from each risk-resolution activity. The overall Boehm's Risk Management activities are shown in Figure 2.4.

Table 2.4 Top 10 Software Risk Items [56]

Risk Item	Risk Management Technique
Personal shortfalls	Staffing with top talent, job matching, team building, and key personnel agreements, cross training.
Unrealistic schedule and budgets	Detailed multisource cost and schedule estimation, design to cost, incremental development, software reuse, requirements scrubbing.
Developing the wrong function and properties	Organization analysis, mission analysis, operations-concept formulation, user surveys and user participation, prototyping, early users' manuals, off-nominal performance analysis, quality-factor analysis.
Developing the wrong user interface	Prototyping, scenarios, task analysis, user participation.
Gold-plating	Requirements scrubbing, prototyping, cost-benefit analysis, designing to cost.
Continuing stream of requirements change	High change threshold, information hiding, incremental development (deferring changes to later increments).
Shortfalls in externally furnished components	Benchmarking, inspections, reference checking, compatibility analysis.
Shortfalls in externally performed tasks	Reference checking, pre-award audits, award-fee contracts, competitive design or prototyping, team-building.
Real-time performance shortfalls	Simulation, benchmarking, modeling, prototyping, instrumentation, tuning.
Straining computer-science capabilities	Technical analysis, cost-benefit analysis, prototyping, reference checking.

Boehm's risk management method together with the Spiral Model for software development provides a comprehensive approach to managing risk for a software project. While this method can be considered as being relatively simplistic, it covers all phases of the software development. This may explain why this method has become the main reference tool and why it is now the most widely used in software project development since its introduction in the 1980s.

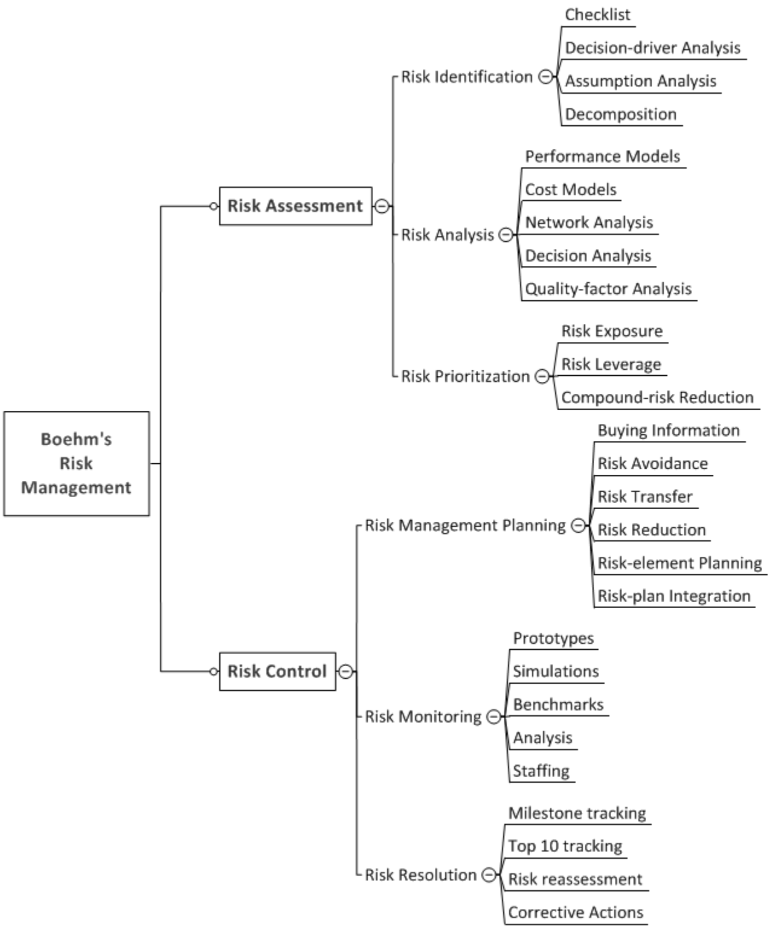


Figure 2.4 Boehm's Risk Management [56]

2.3.3.2 Jones's Risk Management

Caper Jones introduced several methodologies that are used in the areas of software process, software estimation, and software risk management. He identified the ten most influential factors in software project risk [58], which are as follows:

1) Inaccurate metrics. The utilization of Line of Code (LOC) as software metrics introduces errors in software estimation because the amount of LOC in the software will depend on the language and programming style. This type of error will create risk for software planning.

2) Inadequate measurement. Project data collected for project purposes is not always complete and correctly done.

3) Time pressure. The time pressure introduced by management and/or the client creates irrational schedule estimations that create risk in a project. Projects with a high level of complexity and more than 1000 function points (FP) are most likely to have problems with the project schedule.

4) Management weaknesses as well as a lack of knowledge and experience in estimation, planning, measuring and assessment.

5) Inaccuracies in cost estimation.

6) The naive belief that moving to a new technology will create improvements in productivity or quality.

7) Late requirements finalization.

8) Low quality.

9) Low productivity.

10) Cancellation of projects is directly proportional to their size, especially the project with above 10,000 FP or 1 million LOC.

In his most recent papers in software risk management, Jones introduces the Software Risk Master ™, the method for early risk detection via pattern matching followed by risk prevention and risk mitigation [59]. In this method, software project risks are divided into 15 categories such as Health and Safety Risks; Security Risks; Quality Risks; Legal Risks; Traditional Software Risks; Financial Risks; Business Risks; Social Risks; External Risks; Ethical Risks; Knowledge Risks; Enterprise Risks; Merger, Acquisition, and Venture Capital Risks; Technology Risks, and Embedded Software Risks.

Jones' contribution in software risk management focuses mostly on assisting in the risk identification process by providing a list of the typical risks commonly involved in a software project. However, there is no descriptive explanation of how to manage the risks inherent in the overall software project.

2.3.3.3 Karolak's Risk Management

Dale Karolak introduced the Software Engineering Risk Model (SERIM) as a Just-in-Time strategy for risk management in a software project in 1996 [15]. There are six main activities in SERIM to conduct risk management for a software project such as: risk identification, risk strategy and planning, risk assessment, risk mitigation/avoidance, risk reporting, and risk prediction. The relationship between the overall risk management activities is shown in Figure 2.5.

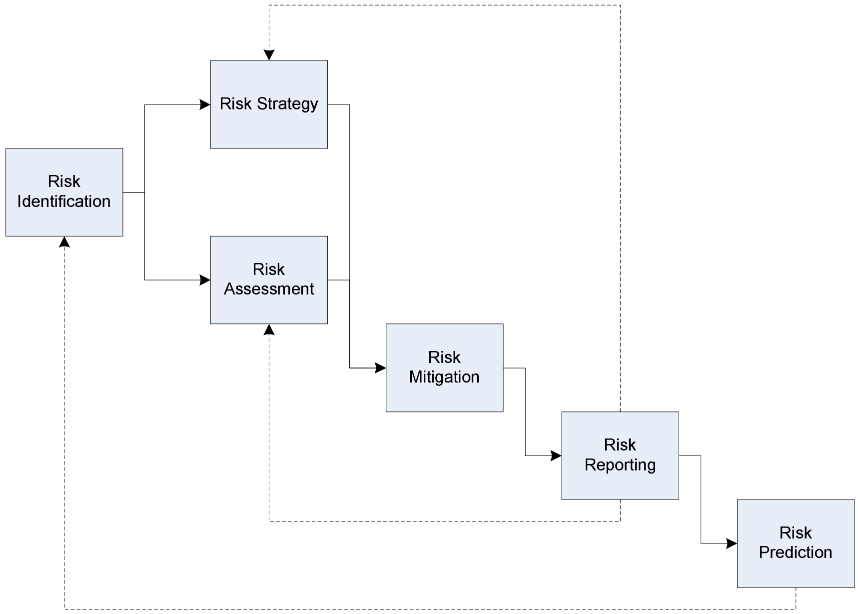


Figure 2.5 SERIM Risk Management Activity Model [60]

In order to have the holistic view of software risk, all activities in SERIM have to be viewed in an operational perspective, a strategic perspective, a technical perspective, a business perspective, an industry perspective, and a practitioner perspective.

The questionnaires that are sent to project stakeholders are key elements in SERIM that are used to identify and measure a high level project risk and to categorize identified risks according to three risk elements: technical risk, cost risk, and schedule risk. These categories are then associated with risk factors, risk metrics and the associated development phases as well as the necessary risk management activities. The overall SERIM risk management model is shown in Figure 2.6.

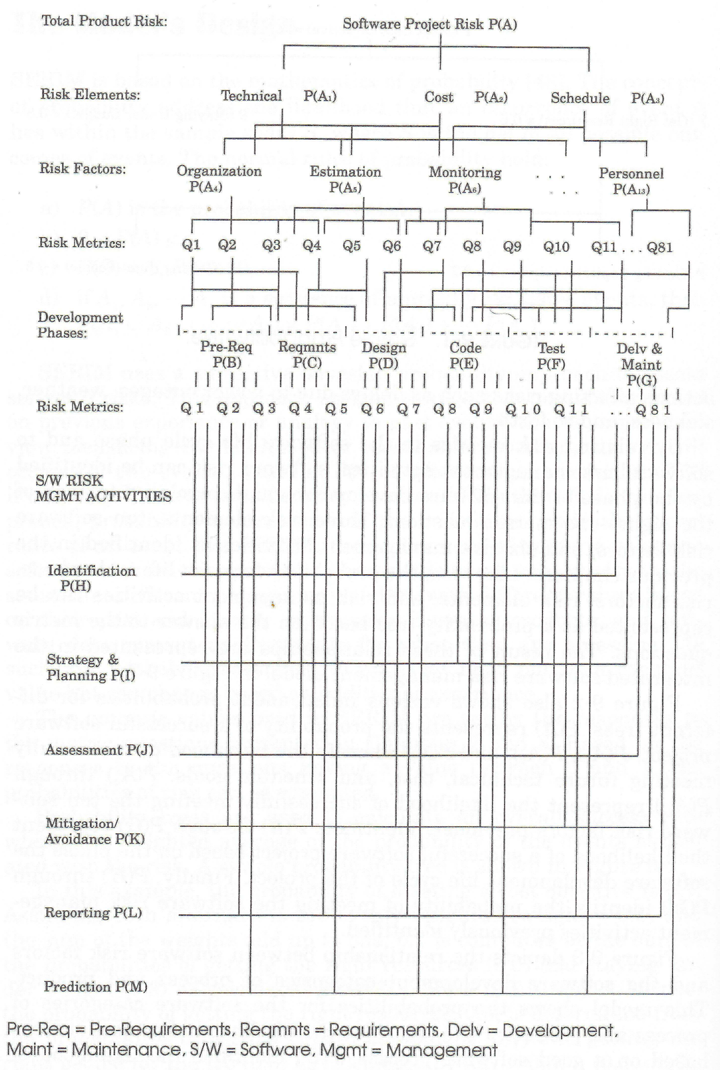


Figure 2.6 Software Engineering Risk Model (SERIM) [61]

The SERIM main contribution is that it provides a simple and flexible way to perform risk management in a software project. However, the method is lacking in explicit guidelines by which to identify risk because a key factor in identifying risk is human experience. As a consequence, the utilization of this method is severely limited even tough its developers claim that it can be used to monitor risk throughout the entire software development life cycle.

2.3.3.4 Charette's Risk Management

Robert Charette introduced the concept of Risk Analysis and Management for software engineering in 1989 [16]. The activities that are involved in this method are: risk identification, risk estimation, risk evaluation, risk planning, risk control, and risk monitoring. The overall activities in risk analysis and management are shown in Figure 2.7.

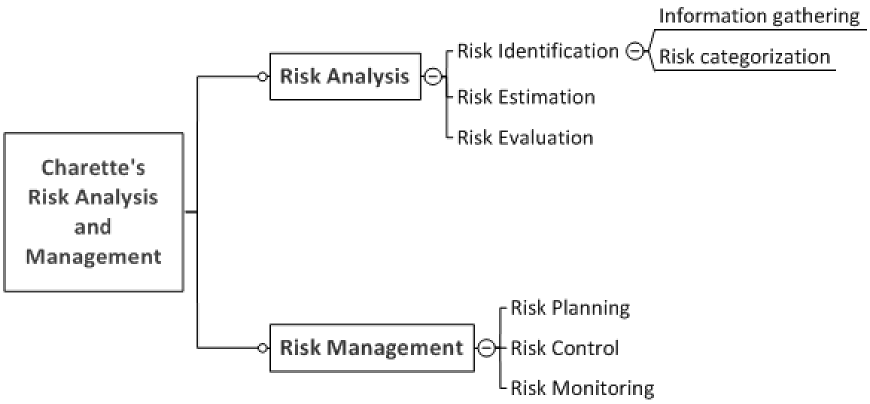


Figure 2.7 Software Risk Analysis and Management [16]

In 1997, Charette described a case study of risk management implementation in the software maintenance phase, which was considered to be more difficult to manage compared to the risks that are inherent in the development phase [62]. This implementation was based on SEI's risk taxonomy, which was used to identify the risks in software maintenance and in SEI's software risk evaluation process to assess the risk. This study showed that risk management in a software project not only deals with the development phase but also takes into consideration the implementation of risk management in the software maintenance phase where risk events are more frequent, come from more diverse sources, and where there is less freedom to respond to the risk. In the implementation stage, Charette's risk management approach relies on human expertise and experience to be successful.

2.3.3.5 Hall’s Risk Management

Elaine Hall defined software risk as “a measure of the likelihood and loss of an unsatisfactory outcome affecting the software project, process, or product” [63]. Here, overall software risk can be placed in three categorizes: software project risk, software process risk, and software product risk.

Software project risk is a management responsibility and is related to the risk in the operational, organizational, and contractual parameters of a project. Software process risk is related to management and technical works procedures, which involve all the project activities inherent in planning, staffing, tracking, quality assurance, and configuration management. Software product risk is a technical responsibility and is related to the end products such as requirement stability, design performance, code complexity, and test specifications. Hall’s software risk classification is shown in Figure 2.8.

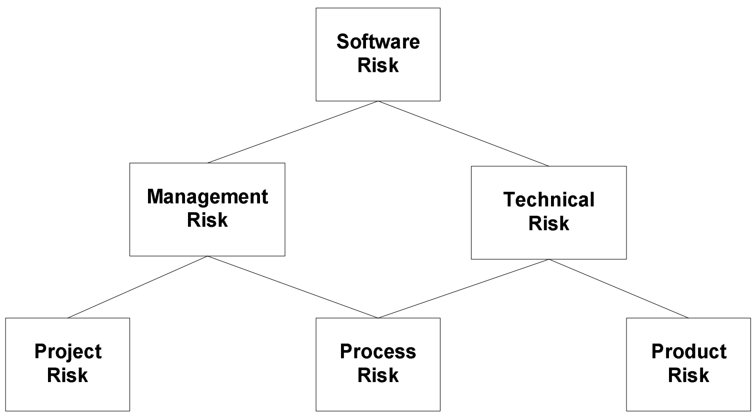


Figure 2.8 Hall’s Risk Classification [64]

Hall’s Risk Management approach suggests that the ability to manage a successful software project, as denoted by the P^2I^2 Success Formula [58], comes from focusing on four main critical success factors: People, Process, Infrastructure, and Implementation.

The people factor is the human resources element in risk management and the most critical factor in managing risk. The people factor influences and drives the other three factors (process, infrastructure, and implementation) in risk management activities. Hence the success of risk management in a software project is highly dependent on successfully managing the people factor.

The process factor in risk management is defined as those activities used to transform uncertainty into acceptable risk. There are five essential elements that must be considered in Hall's approach to risk management, i.e. risk identification, risk analysis, risk planning, risk tracking, risk resolving

The infrastructure factor is defined as those foundation factors used to establish a risk awareness culture in the organization. There are four infrastructure elements that are necessary to take into consideration in performing risk management in the organization, i.e. organization, requirements, resources, and results.

The implementation factor is related to the execution of the risk management plan and the methodology to be applied in a project. The risk management plan consists of mapping the resources necessary to fuel the activities, and the methodology as a set of principles and methods, which include the mechanisms, techniques, and tools that support the risk management implementation. The P^2I^2 Success Formula that becomes the foundation to support the risk management capability of software project organization. Figure 2.9 shows the fish-bone diagram for P^2I^2 Success Formula for risk management.

Hall's risk management formula accommodates other concepts in risk management. These include the utilization of the Software Engineering Institute (SEI) risk taxonomy that is used as a checklist for risk identification and the accommodation of Boehm's risk management process in the Risk Process Factor.

The implementation of this methodology together with all the tools used to identify, analyse, track, and control project risk is highly dependent on human skills and expertise, and as consequences, this methodology is considered complex and difficult to implement [66].

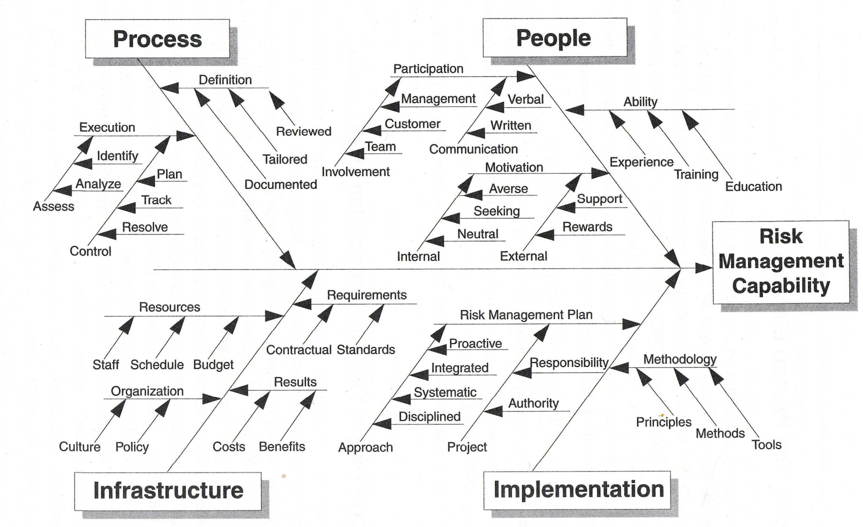


Figure 2.9 P^2I^2 Success Formula in Risk Management [65]

2.3.3.6 SEI's Risk Management

The SEI is a federally funded software engineering research development center sponsored by the US Department of Defense and operated by the Carnegie Melon University in Pittsburgh, Pa, USA. The SEI supports the software engineering community by providing advances on software engineering principles and practices in the area of software acquisition, software security, software development, software process management, software system design, and software risk management [67].

In the area of software risk management, the SEI relies on the basic assumption that a good quality software product is the result of a good quality software development process [68]. This approach depends on the software development process that acts as a means to improve the product's quality and diminish risks.

SEI presents the Software Risk Management (SRM) framework as a holistic vision of the risk-based methodologies for software risk management with the objectives as stated in the SEI documentation as follows [49]:

1. Improve the process of software acquisition in organizations.

2. Improve software risk management methodology, technology, and practice in the acquisition process.

3. Improve the access of software acquisition, repository, use, integration of information and data in industry and government.

4. In general, institutionalize risk management and decision support within the software acquisition community and make it an integral part of the community's practice.

The SRM framework is supported by Three Basic Constructs (Risk Management Paradigm, Risk Taxonomy, Risk Clinic), Three Group of Practices (Software Risk Evaluation (SRE), Continuous Risk Management (CRM), Team Risk Management (TRM)), and Two Methodological Frameworks (Software Capability Maturity Model (SW-CMM^{SM}), Software Acquisition-Capability Maturity Model (SA-CMM^{SM})) as shown in Figure 2.10.

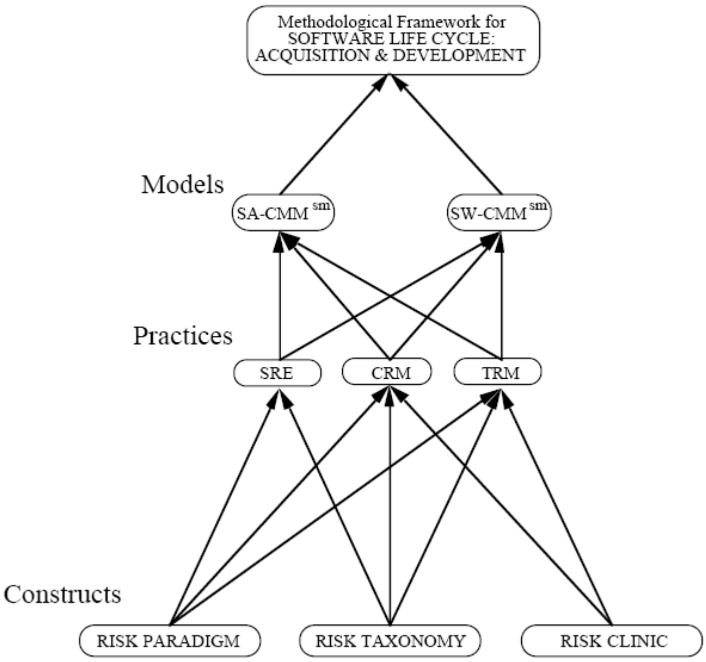


Figure 2.10 SEI Software Risk Management Framework [49]

The SRM framework, which is intended to address risk management in the whole software lifecycle starting from software acquisition through its development, and maintenance by means of the SEI Risk Management Paradigm. The paradigm is depicted as a continuous risk management activity starting with risk identification and going through risk analysis, risk planning, risk tracking, and risk controlling all in the project lifecycle as shown in Figure 2.11.

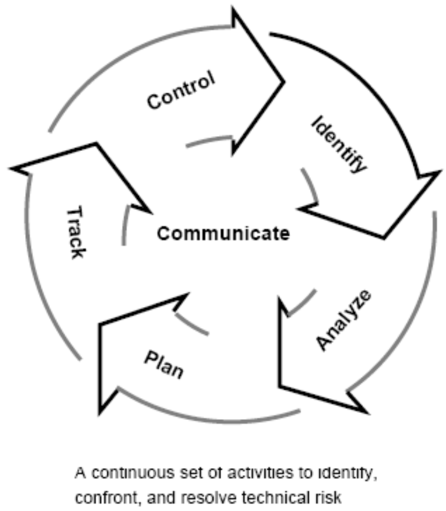


Figure 2.11 SEI Risk Management Paradigm [49]

The risk identification activity uses Taxonomy based Questionnaire (TBQ), which is based on the risk taxonomy used to search and locate risks before they become a problem. The SEI Risk taxonomy organizes the risks into class level, element level, and attributes level as show in Figure 2.12.

The risk analysis activity is focused on creating meaningful information from risk data by building and evaluating the risk model. The information created from this activity will in turn be used for decision-making purposes.

Risk planning activities convert the decision-making information into plans and actions. The activities in this phase include the planning for risk response and the acquisition of further information concerning the nature of a risk, where more information is needed on which to base subsequent decisions.

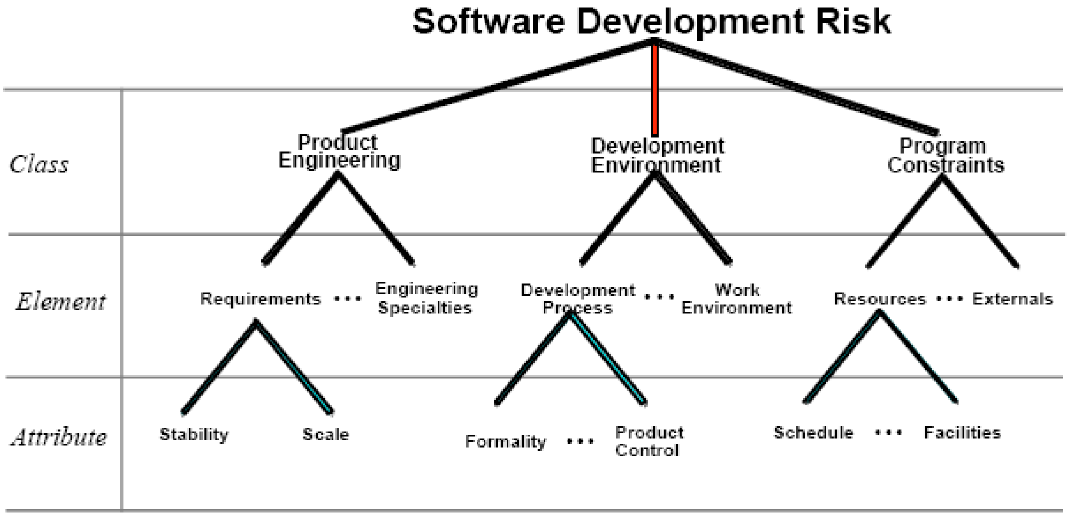


Figure 2.12 SEI Software Risk Taxonomy [49]

Risk Tracking identifies and monitors the project risks through the trigger events and also monitors the risk response actions that have been taken for every risk.

Risk Control is the last activity in the loop, which consists of correcting for deviations from planned actions. The central point of all the closed loop risk activities defined by SEI is communication, which is the main activity for successful risk management in a software project. The detailed SEI risk taxonomy is shown in Table 2.5.

The SEI-SRM provides a comprehensive approach to risk management, which is complete with a well defined interview to be conducted for risk identification. However this approach relies heavily on human experience for the conducting of risk management activities and risk taxonomy. This approach also highly subjective and difficult to measure such as the difficulties to measure politics and morale [66][69].

Table 2.5 Taxonomy of Software Risks: Overview [49]

A. Product Engineering	B. Development Engineering	C. Program Constrains
1. Re**quirements**	**1. Development Process**	**1. Resources**
a. Stability b. Completeness c. Clarity d. Validity e. Feasibility f. Precedent g. Scale	a. Formality b. Suitability c. Process Control d. Familiarity e. Product Control	a. Schedule b. Staff c. Budget d. Facilities
2. **Design**	**2. Development System**	**2. Contract**
a. Functionality b. Difficulty c. Interfaces d. Performance e. Testability f. Hardware Constraints g. Non-Developmental Software	a. Capacity b. Suitability c. Usability d. Reliability e. System Support f. Deliverability	a. Type of Contract b. Restrictions c. Dependencies
3. **Code and Unit Test**	**3. Management Process**	**3. Program Interface**
a. Feasibility b. Testing c. Coding/Implementation	a. Planning b. Project Organization c. Management Experience d. Program Interface	a. Customer b. Associate Contractors c. Subcontractors d. Prime Contractor e. Corporate Management f. Vendors g. Politics
3. **Integration and Test**	**4. Management Methods**	
a. Environment b. Product c. System	a. Monitoring b. Personnel Management c. Quality Assurance d. Configuration Management	
5. **Engineering Specialties**	**5. Work Environment**	
a. Maintainability b. Reliability c. Safety d. Security e. Human Factors f. Specifications	a. Quality Attitude b. Cooperation c. Communication d. Morale	

2.4 Expert-COCOMO Model

The Expert COCOMO Model was introduced by Ray Madachy with the primary aim to detect and analyze input anomalies for project effort estimation [19]. The model is an extension of COCOMO and is also known as the Heuristic Risk Assessment Model that can be used in the project planning phase for identifying, categorizing, and prioritizing project risks.

Several risk management models for software projects depend largely on human expertise in their application especially in the risk identification and analysis phases. This dependency has become the main barrier to the implementation of risk management in software projects because a lot of time was required to do a thorough analysis, and because it was perceived as being both effort intensive and costly [19][20].

Even though the risk assessment (identification and analysis) is conducted together with the software estimation in the Project Planning phase, risk identification and analysis is usually done separately from cost estimation. The Expert-COCOMO model improves on this process by utilizing the information taken from the effort estimation step to establish a risk assessment for a particular software project.

The cost factors in the form of scale factors and effort multipliers in the COCOMO model become the inputs for the Expert-COCOMO model. The output for the model is a list of software risks that are related to the COCOMO cost factors, such as: *Schedule Risk*, *Product Risk*, *Platform Risk*, *Personnel Risk*, *Process Risk,* and *Reuse Risk.* The software risk taxonomy in the Expert-COCOMO model is described in Figure 2.13.

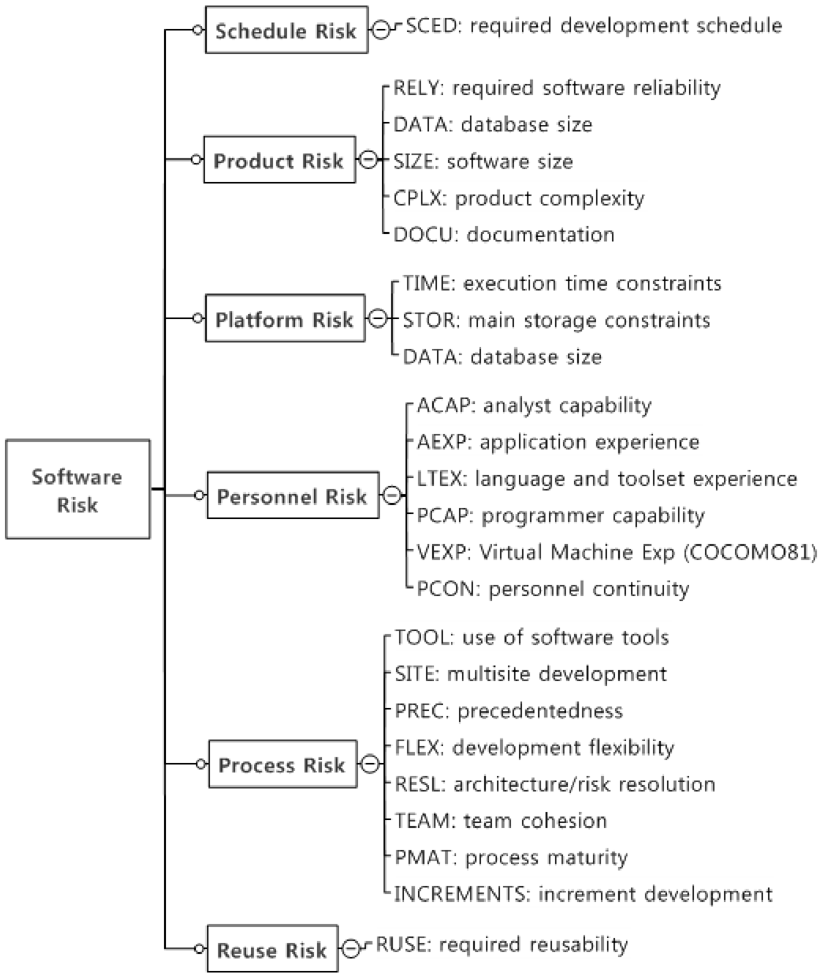


Figure 2.13 Expert-COCOMO Risk Taxonomy [19]

All risks in the Expert-COCOMO model, which are defined as the result of a combination of several cost factors and risk rules used to determine the level of every risk, can be identified by mapping 2 cost factors (attributes) according to a risk level assignment matrix as shown in Figure 2.14.

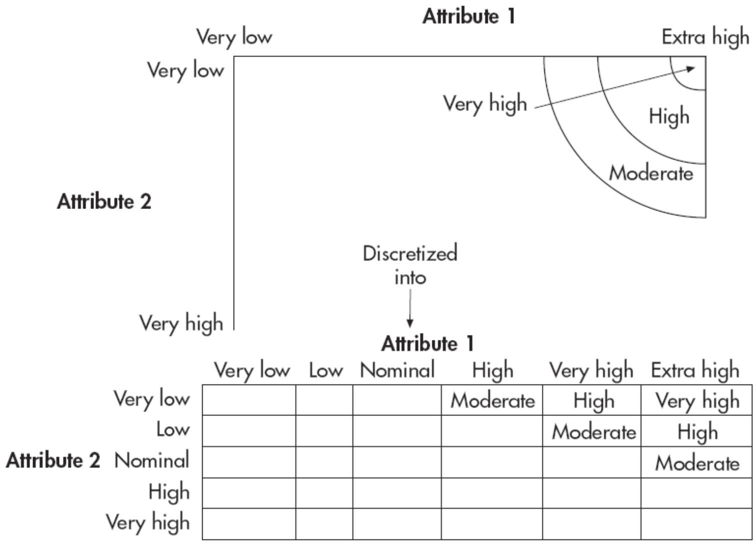


Figure 2.14 Risk Level Assignment Matrix [19]

Schedule Risk will emerge if a project with a tight schedule is being developed by a developer(s) with low technical capability. Schedule risk is also considered to be high for a project within which the manager attempts to develop a complex product within a tight schedule.

Product Risk is a software project risk that is related to the software product as the process deliverable, which is affected by such product related parameters as: required reliability of software product, product size, and product complexity as well as database and documentation requirements.

Platform Risk is related to the volatility of the development platform that could introduce many problems in the future and the necessity to rework certain project steps.

Personnel Risk is the primary source of project risks and affects the overall productivity of a software project. Personnel risk is related to the analyst's overall capabilities and experience as well as their specific experience with the programming language and tools being used. It also depends on the capabilities of the programmer involved in the project. The combination of low programmer capability and a tight project schedule will create a significant risk for the project.

Process Risk is that aspect of risk related to the project attribute category in COCOMO known as software tools, multisite development, precedentedness, development flexibility, architecture resolution, team cohesion, and process maturity.

Reuse Risk is related to the impact of using a reuse application in software development. Reuse risk will depend on the reuse strategy that requires reliability, experience, appropriate tools, and other elements to ensure the success of a product.

Overall Project Risk quantifies the level of risk as it relates to the combination of cost factors in a software project. The nonlinearity of the assigned risk levels and the cost multiplier data are used to compute the overall risks for each category and for the entire project according to the equation (2.2), where the effort multiplier product = (driver 1 effort multiplier) x (driver 2 effort multiplier) x . . . x (driver *n* effort multiplier) [19].

$$\textbf{Project Risk} = \sum_{j=1}^{\#categories} \sum_{i=1}^{\#category\ \ risks} \textbf{risk level}_{ij} \times \textbf{effort multiplier product}_{ij} \quad \text{…. (2.2)}$$

Based on the above formula, the overall project risk can be defined as follows:

Table 2.6 Project Risk Category

Project Risk	Risk Category
0 - 15	Low
5 - 15	Moderate
15 - 50	High
50 - 100	Very High

A recent Expert-COCOMO application was developed using a C programming language and an HTML interface and is posted at the USC website [70] as shown in Figure 2.15.

A recent Expert-COCOMO application was developed using a C programming language and an HTML interface and is posted at the USC website [70] as shown in Figure 2.15. The output of this application is an estimated effort based on COCOMO-II and the risk assessment summary, which consists of an overview of the total project and a risk schedule giving details of the product risk, platform risk, personnel risk, process risk, reuse risk, and all the other individual risk items.

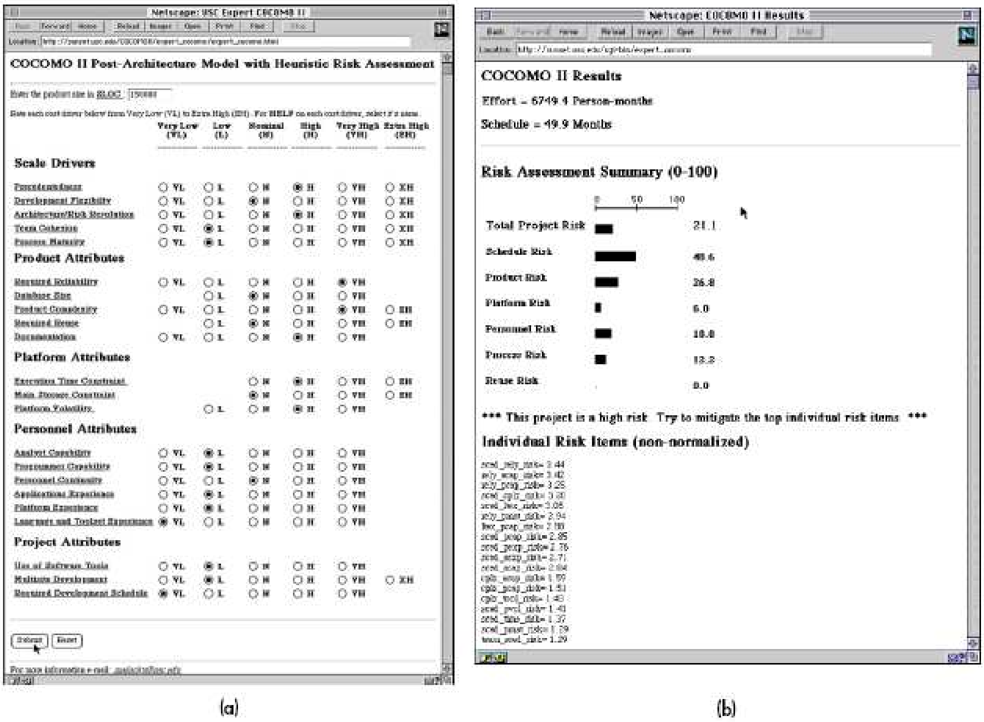


Figure 2.15 (a) Sample input screen (b) Sample risk outputs [19]

Compare to other approaches, Expert-COCOMO provides the efficient approach to software project risk management especially in the early risk identification and analysis activities [19]. This approach enables the early risk identification and risk analysis activity to be done in the project planning phase together with software estimation activity. This approach leverages the existing knowledge and expertise taken from effort estimation activities to assess the level of risk in a new software development project. Expert-COCOMO provides a list of risks and priorities that will help the project manager develop a detailed risk analysis and create a risk mitigation plan.

However, this model has some limitations because it cannot effectively deal with imprecise and uncertain information inputs in the form of linguistic terms such as: Very Low (VL), Low (L), Nominal (N), High (H), Very High (VH) and Extra High (XH). The difficulties associated with defining risk more accurately for a deeper analysis also considerably worsen, due to the fact that risk is an abstract and fuzzy concept [24].

2.5 Fuzzy Logic

The Fuzzy Logic System is one of three main components of soft computing, the field in computer science that deals with imprecision, uncertainty, and approximation to achieve practicability, robustness and low cost solutions. The soft computing components, such as: Neural Network Theory, Probabilistic Reasoning, and Fuzzy Logic; perform the techniques that mimic the ability of the human mind to deal with reasoning and approximation problems rather than those that are more exact [71].

If a Neural Network is primarily concerned with learning ability and Probabilistic Reasoning deals with uncertainty, then the Fuzzy Logic methodology introduced by Prof. Lofti Zadeh in 1965 provides a useful tool for dealing with imprecision, uncertainty, and complexity in problems that are difficult to solve quantitatively [72].

The Fuzzy Logic System deals with fuzzy parameters, which address imprecision and uncertainties, by mapping out the path of a given input to an output using the computing framework called the Fuzzy Inference System [73].

This framework consists of three main processes: the Fuzzification Process, the Inferences Process from Fuzzy Rules, and the Defuzzification Process [74]. Figure 2.16 is a diagram of the fuzzy inference system with the three main processes.

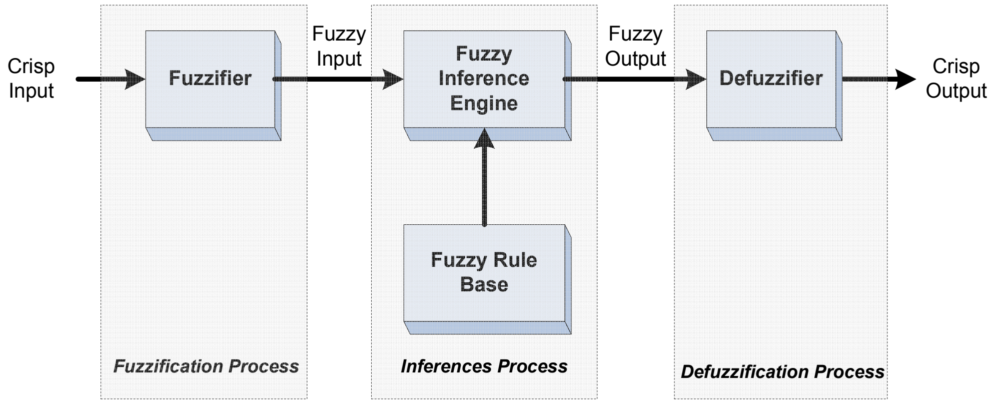


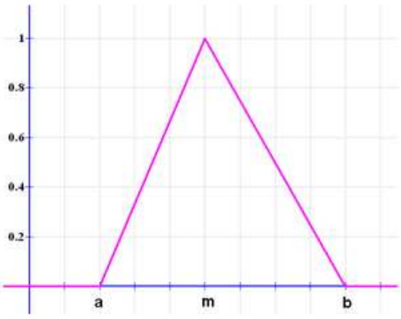
Figure 2.16 Fuzzy Inference System [74]

2.5.1 Fuzzification Process

The Fuzzification Process consists of a fuzzifier that transforms crisp input into a fuzzy set of values based on its membership function (MF). A fuzzy set is a mathematical model comprised of vague qualitative or quantitative data, which is frequently generated by means of the natural language. The membership function is a curve that maps the inputs to a membership value that ranges between 0 and 1.

The fuzzification process allows the input to the system to be expressed in linguistic terms. The most popular membership functions in a fuzzy system are triangular, trapezoidal, and Gaussian.

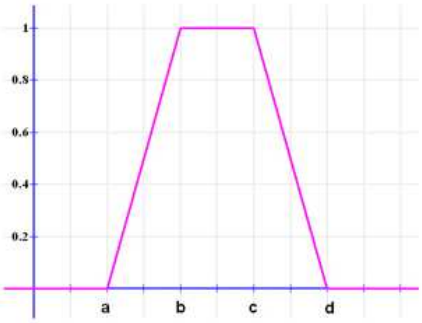
Triangular MF is a function of $\mu_A(x)$ that depends on the scalar parameter a as a lower limit, b as an upper limit, and $a < m < b$. Figure 2.17 shows the diagram and formula of Triangular MF.



$$\mu_A(x) = \begin{cases} 0, & x \leq a \\ \dfrac{x-a}{m-a}, & a < x \leq m \\ \dfrac{b-x}{b-m}, & m < x < b \\ 0, & x \geq b \end{cases}$$

Figure 2.17 Triangular Membership Function

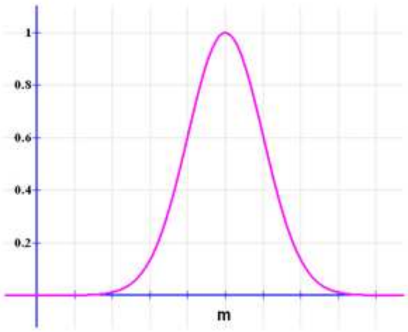
Trapezoidal MF is a function of $\boldsymbol{\mu}_A(x)$ that depends on a lower limit $\boldsymbol{a}$, an upper limit $\boldsymbol{d}$, a lower support limit $\boldsymbol{b}$, and an upper support limit $\boldsymbol{c}$, where $\boldsymbol{a < b < c < d}$**.** Figure 2.18 shows the diagram and formula of Trapezoidal MF.



$$\mu_A(x) = \begin{cases} 0, & (x < a) \text{ or } (x > d) \\ \dfrac{x-a}{b-a}, & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \dfrac{d-x}{d-c}, & c \leq x \leq d \end{cases}$$

Figure 2.18 Trapezoidal Membership Function

Gaussian MF is a function of $\boldsymbol{\mu}_A(x)$ that depends on a central value $\boldsymbol{m}$ and a standard deviation $\boldsymbol{k > 0}$. The smaller k is, the narrower the “bell”. Figure 2.19 shows the diagram and formula of Gaussian MF.



$$\mu_A(x) = e^{-\frac{(x-m)^2}{2k^2}}$$

Figure 2.19 Gaussian Membership Function

2.5.2 Inference Process

The inference process involves the fuzzy inference engine that is used to perform the mapping between the input from the fuzzification process and the output based on expert knowledge or rules. The role of fuzzy rules in the inference process is to capture the imprecise modes of reasoning and to act as the means to produce the fuzzy output from the fuzzy input.

A fuzzy rule is also known as the Fuzzy IF-THEN rule and is generally expressed as follows [75]:

$$\text{IF } (\boldsymbol{x} \textit{ is } \boldsymbol{A}) \text{ AND } (\boldsymbol{y} \textit{ is } \boldsymbol{B}) \text{ THEN } (\boldsymbol{z} \textit{ is } \boldsymbol{Z}) \qquad \ldots (2.3)$$

Where

x, y, z represent the variables, and

A, B, Z are the linguistic values in the universe of discourse.

This rule can be divided into two parts, the ***IF*** part, which is referred to as the *antecedent* or *premise* that contains the fuzzy description of the measured input values, and the ***THEN*** part, which is referred to as the *consequent* or *conclusion* that defines a possible fuzzy output for every corresponding input.

In this process, several fuzzy rules can be applied in the fuzzy inference engine to build the knowledge base to be used for decision-making by performing an aggregate of fuzzy operations to map the path of a given input to an output, which is based on the generalized modus ponens as follows [74]:

Premise 1: IF ***x*** is ***A*** THEN ***y*** is ***B***

Premise 2: ***x*** is ***A'*** … (2.4)

Conclusion: ***y*** is ***B'***

Where ***A***, ***A'***, ***B***, ***B'*** are fuzzy sets and ***x*** and ***y*** are symbols named for objects.

The inference process creates the fuzzy output as the aggregation from several fuzzy rules.

Figure 2.20 illustrates the inference process, which involves three fuzzy rules with three parameters i.e. service, food, and tip.

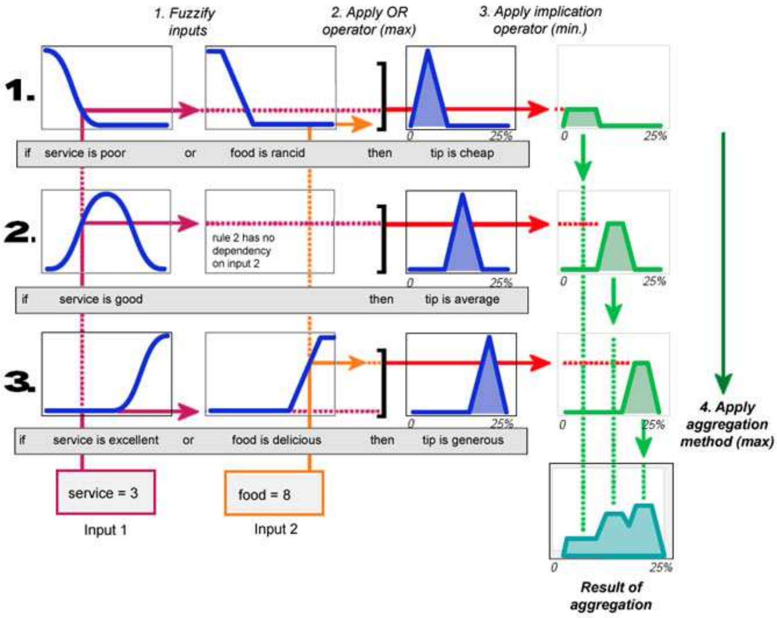


Figure 2.20 Fuzzy Inference Process [76]

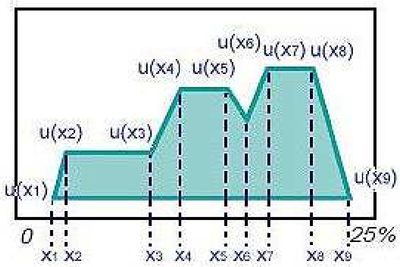
2.5.3 Defuzzification Process

The defuzzification process produces and translates an aggregate fuzzy output from the inference process into a quantifiable result or crisp output. The most popular defuzzification method is the centroid calculation, which returns the center of an area under the curve according to Formula 2.5.

$$x^* = \frac{\sum_{i=1}^{n} x_i \cdot \mu_A(x_i)}{\sum_{i=1}^{n} \mu_A(x_i)} \qquad \text{... (2.5)}$$

Where $\boldsymbol{n}$ is the number of discrete elements, $\boldsymbol{x_i}$ is the value of the discrete element, and $\boldsymbol{\mu}_{\mathrm{A}}$ $(\boldsymbol{x_i})$ represent the corresponding MF value at the point $\boldsymbol{x_i}$.

Figure 2.21 illustrate the defuzzification process to transform nine fuzzy values into one crisp output.



$$g = \frac{\sum_{i=1}^{9} x_i \cdot u(x_i)}{\sum_{i=1}^{9} u(x_i)}$$

Figure 2.21 Defuzzification Process [75]

With the ability to mimic the human mind and to deal with uncertainty, the fuzzy logic system provides the right tool for risk management that involves imprecision in the form of the likely outcome from an uncertain event. Fuzzy logic is also capable of improving on the Expert-COCOMO model by accommodating the risk calculation model that is based on input from various cost factors, which are described in the form of such uncertain parameters as: Very Low (VL), Low (L), Nominal (N), High (H), Very High (H), and Extra High (EH).

2.6 Chapter Summary

This chapter describes the background theories related to the problems identified in the research questions and can be summarized as follow:

- Effort estimation and risk management are the primary elements in software project planning phase that must be execute together in order to increase the chance of success in managing a software project. However, effort estimation and risk management activities in the most of software project planning are often disconnected with each other.

- Every software effort estimation methodologies including COCOMO provides the fix estimation value and do not provide contingency allowance for their estimation value to cover the risks and assumption.

- The contingency allowance is highly related to software risk and the ability in providing the contingency allowance value is highly dependent on the ability to conduct the software risk assessment as the integral part of effort estimation activity.

- Several risk management models for software projects was perceived as being both effort intensive and costly because it depends largely on human expertise and experience. This dependency has become the main barrier to the implementation of risk management in software projects.

- Expert-COCOMO is the risk assessment based on cost factors that provides the efficient approach to software project risk management especially in the early risk identification and analysis activities. This approach leverages the effort estimation activities to assess the level of risk in a new software development project and could be done in the project planning phase together with software estimation activity. However, this model has some limitations because it cannot effectively deal with imprecise and uncertain information inputs.

Chapter 3
Fuzzy Expert-COCOMO Model

From the discussion in the previous chapter we understand that Effort Estimation and Risk Assessment are an integral part of Project Planning Phase and play a key role in ensuring the success of a software development project. This can be understood since the accuracy of the effort estimation is highly dependent on the size, nature, and number of project risks, which are inherent in a particular software project. However, in the common practice, these two activities are often disconnected from each other. Moreover, the existing methodologies in effort estimation and risk assessment also have some weaknesses.

The main weakness with respect to the existing effort estimation methodology is that it is only able to provide an exact value rather than an based-value and allowance value for the software development effort. On the other hand, because the existing risk assessment methodology is highly dependent on the human judgement and expertise, it is very expensive to implement. Hence, while the Expert-COCOMO could be considered to be the efficient method for early risk assessment, it has a weakness with respect to dealing with the uncertain parameter of the inputs.

This chapter describes the Fuzzy-ExCOM (Fuzzy Expert-COCOMO) Model as being an improvement on the Expert-COCOMO model with the advantage that it provides for more sensitive information input about software project risks and the ability to integrate the risk assessment and effort estimation activities by calculating the contingency allowance for effort estimation based on identified project risks.

This chapter is organized as follows: Section 3.1 provides an overview of the Fuzzy-ExCOM model, the Risk Model is described in Section 3.2, and Section 3.3 describes the Effort Contingency Model.

3.1 Fuzzy-ExCOM Model

Fuzzy-ExCOM Model is the integrated risk assessment and effort contingency model based on fuzzy logic that provides the effort estimation improvement based on identified project risk. Fuzzy logic is applied to the model because of its capability in modeling complex system with imprecise parameters and has proved to be very successful in many fields, such as control system, decision support system, and other expert systems [77].

Fuzzy-ExCOM consists of two models, Risk Model and Effort Contingency Model that provides the complementary information for effort estimation value. These models identified potential project risks based on cost factors and calculate the effort contingency value that should be prepared in order to accommodate the identified risks. The overall diagram of Fuzzy-ExCOM Model is shown in Figure 3.1.

3.2 Risk Model

As discussed in the previous chapter, the main limitation of the existing Expert-COCOMO approach is the difficulties in defining the accurate risk in the identification process because the inputs are in the form of linguistic term such as: Very Low (VL), Low (L), Nominal (N), High (H), Very High (VH) and Extra High (XH).

The Risk Model is software risk assessment methodology based on fuzzy-logic and Expert-COCOMO. Fuzzy logic improves the sensitivity of risk identification in Expert-COCOMO and is applied to the cost factor parameters as the input for Expert-COCOMO that usually describes the qualitative measurements such as very low, low, nominal, high, and very high.

There are 22 inputs for the model which consist of 5 scale factors and 17 cost drivers, and 7 outputs, which are schedule risk, personnel risk, process risk, product risk, platform risk, reuse risk, and project risk.

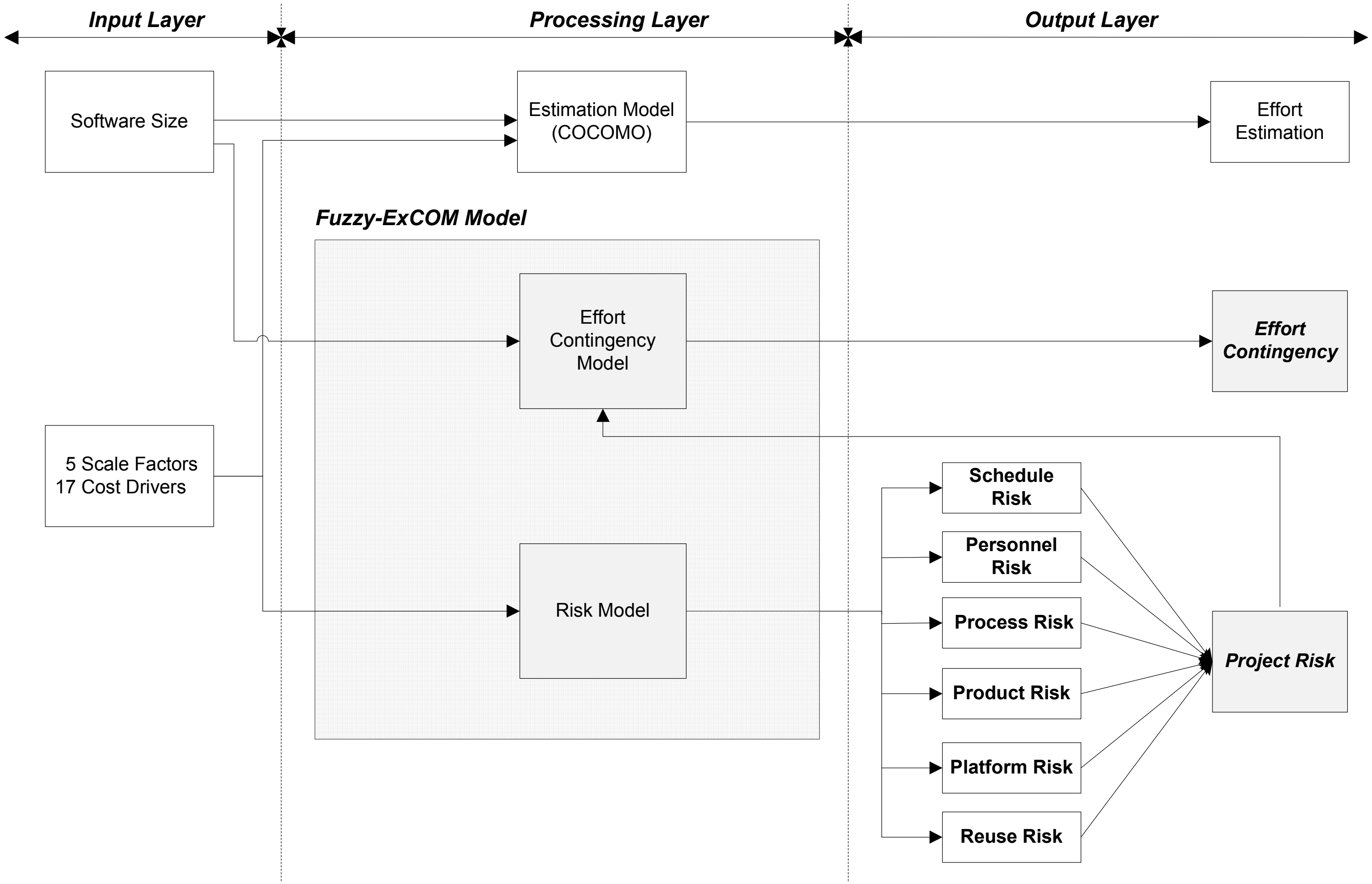


Figure 3.1 Fuzzy-ExCOM Diagram

Fuzzy Logic calculates the software risks in Fuzzy-ExCOM Risk Model through the following processes: a cost factor fuzzification process, a risk level assignment based on fuzzy inference rules, a risk level defuzzification process, and risk quantification. Figure 3.2 shows the diagram of Fuzzy-ExCOM Risk Model.

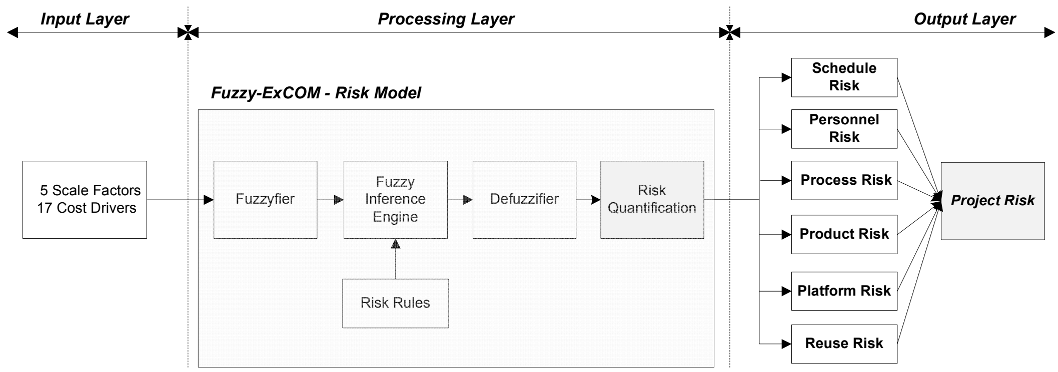


Figure 3.2 Fuzzy-ExCOM Risk Model

3.2.1 Fuzzification Process

The fuzzification process converts the qualitative input of 5 scale factors and 12 cost drivers value to the fuzzy value based on cost factors value listed in Table 2.3 at chapter 2. Gaussian membership function is used as the input conversion function due to the characteristic of this membership function as the most adequate approach to represent uncertainty parameters [78].

3.2.2 Fuzzy Inference Engine and Risk Rules

The Fuzzy Inference Engine determines the level of every risk based on the risk rules that are created from 2 input parameters and a risk level assignment matrix as shown in Figure 2.14 at chapter 2.

There are 31 risk rules created in Fuzzy-ExCOM based on Expert-COCOMO's risk rules which divides in the 6 risk categories as follow:

1. Schedule Risk

   Main parameter: *SCED*

   Rules: *sced_cplx; sced_rely: sced_time: sced_pvol; sced_tool; sced_acap: sced_aexp; sced_pcap; sced_ltex; sced_pmat; sced_team*

2. Personnel Risk

   Main parameters: *ACAP; AEXP; LTEX; PCAP*

   Rules: sced_acap; *pmat_acap; tool_acap; rely_acap; cplx_acap; time_acap; stor_acap; sced_aexp; ruse_aexp; team_aexp; sced_ltex; ruse_ltex; sced_pcap; pmat_pcap; tool_pcap; rely_pcap; cplx_pcap; time_pcap; stor_pcap*

3. Process Risk

   Main parameters: *TOOL; SITE; TEAM; PMAT*

   Rules: *sced_tool; tool_acap; tool_pcap; tool_pmat; cplx_tool; time_tool; team_site; sced_pmat; pmat_acap; pmat_pcap; tool_pmat; rely_pmat*

4. Product Risk

   Main parameters: *RELY; CPLX*

   Rules: *sced_rely; rely_acap; rely_pcap; rely_pmat; sced_cplx; cplx_acap; cplx_pcap; cplx_tool*

5. Platform Risk

   Main parameters: *TIME; STOR; PVOL*

Rules: *sced_time; time_pcap; time_acap; time_tool; stor_acap; stor_pcap; sced_pvol*

6. Reuse Risk

   Main parameters: *RUSE*

   Rules: *ruse_aexp; ruse_ltex*

The overall diagram of risk rules and their categorization to the project risk is shown in Figure 3.3.

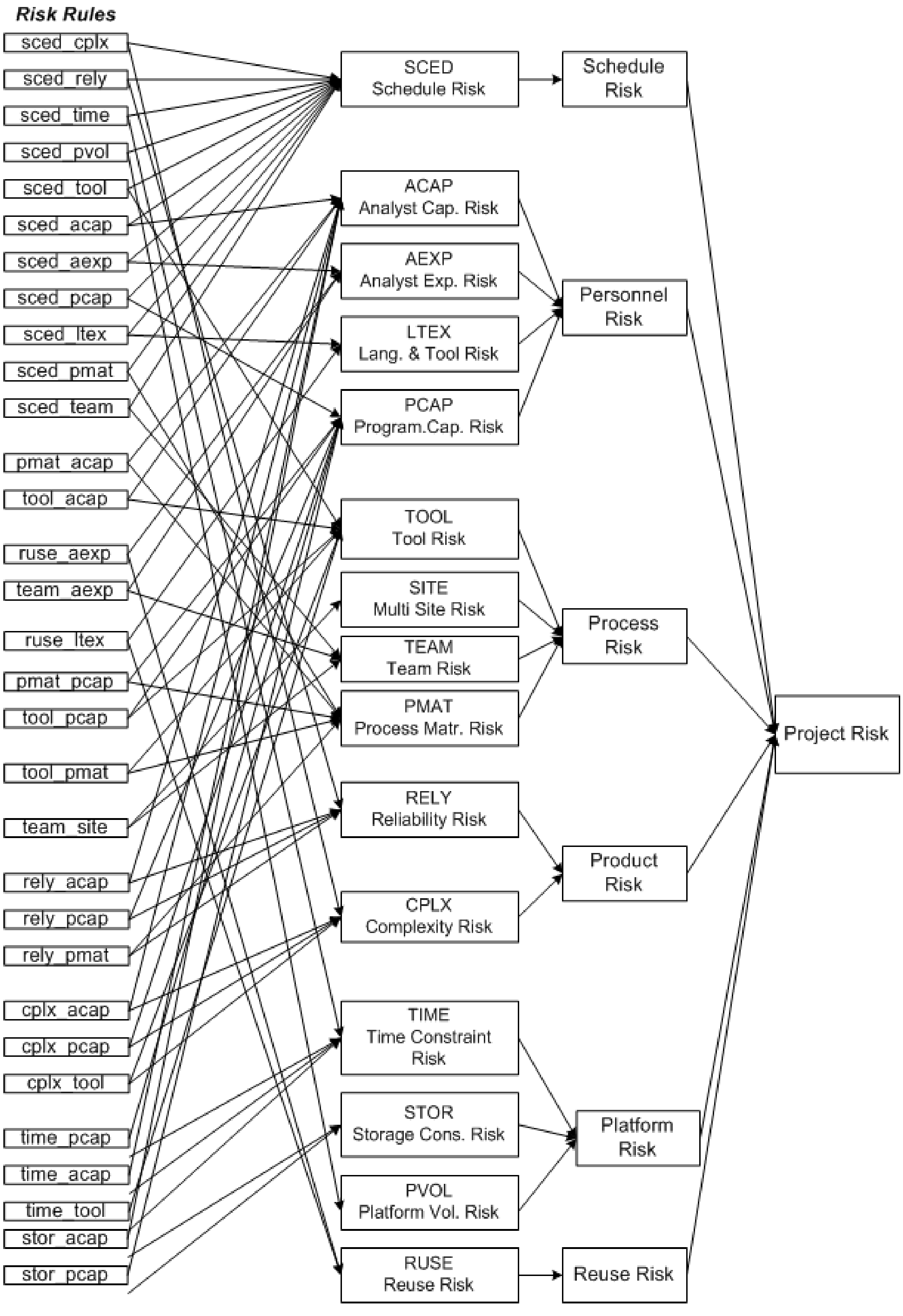


Figure 3.3 Risk Rules

Figure 3.4 shows the risk rules matrices for SCED parameter. There are 11 risk rules that related to SCED. Risk rules related to other cost factors describe mode detail at Appendix C.

CPLX \ SCED	Very Low	Low	Nominal	High	Very High
Very Low					
Low	Very Low				
Nominal	Low	Very Low			
High	Moderate	Low	Very Low		
Very High	High	Moderate	Low	Very Low	
Extra High	Very High	High	Moderate	Low	Very Low

AEXP \ SCED	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				

RELY \ SCED	Very Low	Low	Nominal	High	Very High
Very Low	Very Low				
Low	Low	Very Low			
Nominal	Moderate	Low	Very Low		
High	High	Moderate	Low	Very Low	
Very High	Very High	High	Moderate	Low	Very Low

PCAP \ SCED	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				

TIME \ SCED	Very Low	Low	Nominal	High	Very High
Nominal	Very High	High	Moderate	Low	Very Low
High	High	Moderate	Low	Very Low	
Very High	Moderate	Low	Very Low		
Extra High	Low	Very Low			

LTEX \ SCED	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				

PVOL \ SCED	Very Low	Low	Nominal	High	Very High
Low	Low	Very Low			
Nominal	Moderate	Low	Very Low		
High	High	Moderate	Low	Very Low	
Very High	Very High	High	Moderate	Low	Very Low

PMAT \ SCED	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				
Extra High					

TOOL \ SCED	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				

TEAM \ SCED	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				
Extra High					

ACAP \ SCED	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				

Figure 3.4 SCED Risk Rules

3.2.3 Defuzzification and Risk Quantification Process

The defuzzification process transforms results from inference process in the fuzzy form to crisp results for every risk. The quantification process calculates software project risks based on Formula 2.2 and categorizes as Low Risk Project, Moderate Risk Project, High Risk Project, and Very High Risk Project based on Table 2.6 in Chapter 2.

3.3 Effort Contingency Model

Fuzzy-ExCOM Effort Contingency Model calculates the contingency allowance for COCOMO effort estimation based on project risk and software size. In software development project, the higher risk can be understood that the project will have high probability of unintended event that can affect the project cost, time, and quality. This means the project manager should prepare higher contingency allowance for project with higher risk because the contingency allowance is proportional to the project risk [79]. Another parameter that should also be considered in the contingency allowance calculation is software size. The project that developed the bigger software size compared to the other project will have a higher uncertainty level (risk) compared to the project for smaller software size [59].

Fuzzy-ExCOM Effort Contingency model provides the information about contingency allowance based on project risk and software size in addition to the effort estimation value. The model provides the integrated approach in software project planning, start from effort estimation, risk assessment, and contingency allowance. Fuzzy logic implemented to the model in accommodating the effort contingency calculation which involves the uncertain values such as risk. Figure 3.5 shows the overall Fuzzy-ExCOM Contingency model.

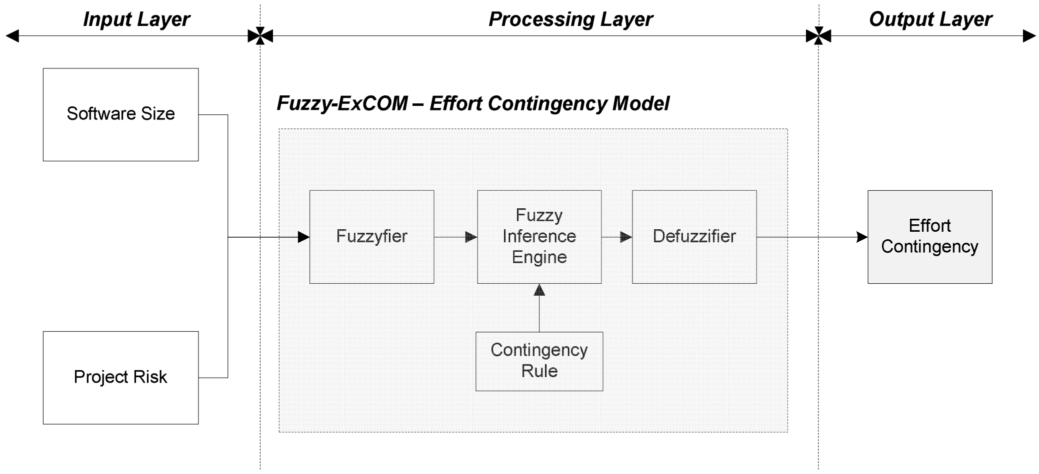


Figure 3.5 Fuzzy-ExCOM Effort Contingency Model

Effort Contingency Model consists of 3 fuzzy processes to calculate the effort-contingency, they are fuzzification process, fuzzy inference process, and defuzzification process.

Fuzzifier in the fuzzification process transforms the inputs in the forms of software size and software project risk to the fuzzy set value. The software size categorization based on the definition that a large system software project is about 10,000 function points or greater [59] or about 128 KLOC, while a Super Large System was taken to be 512 KLOC or more [80]. Software size categorization in this research is shown in Table 3.1.

Table 3.1 Software Size Categorization

Size	Value (KLOC)
Small	0 – 50.0
Medium	50.1 – 128.0
Large	128.1 – 512.0
Extra Large	512.1 – up

The contingency rule in the inference process calculates the contingency value based on the combination matrix between Software Size and Software Project Risk. In this rule, the low risk project which develops small size software will need low contingency, while high risk project which develops large size software will required high contingency value. The overall rules in contingency model are shown in Table 3.2.

Table 3.2 Contingency Rule

Project Risk	Software Size: Small	Medium	Large	X-Large
Low	Low	Low	Medium	Medium
Moderate	Low	Medium	High	High
High	Medium	High	High	V High
Very High	Medium	High	V High	V High

In the defuzzification process, the fuzzy value of contingency allowance as output of inference process will transform to crisp value. Contingency value describes the percentage amount that should be added to effort estimation base-value. Barry Boehm mentioned that the software estimation would be good with the 20% accuracy of cost and 70% accuracy of time [81]. Based on the above range the contingency allowance value is defined between the values of 0% and 100%. The contingency allowance is categorized as Low, Medium, High, and Very High. The overall value of Effort Contingency in Fuzzy ExCOM is shown in Table 3.3.

Table 3.3 Effort Contingency Value

Contingency	Value
Low	0% - 25.0%
Medium	25.1% –50.0%
High	50.1% – 75.0%
Very High	75.1% – 100%

Contingency allowance provides a range of values for COCOMO Effort Estimation instead of fixed value. The new effort estimation value will be in form of base-value, minimum value, and maximum value. Software project with effort estimation base of 100 person-month and contingency allowance of 25% can be described as having the 100 person-month as base-value with minimum value of 75 person-month and maximum value of 125 person-month.

Contingency allowance provides the better and meaningful estimation value for project planning purposes, because the allowance value accommodates the project risks and estimation assumption which is not covered in original estimation methodology.

3.4 Chapter Summary

This chapter introduces a new model called by Fuzzy-ExCOM Model which has the following characteristics and capabilities:

- Improved the risk assessment results using Expert-COCOMO by utilizing the fuzzy techniques to overcome inputs in the form of linguistic terms.
- Improved project planning process by integrating the Effort Estimation activity and Risk Assessment activity in software development project.
- Improved effort estimation results by providing Effort Contingency Allowance that is based on software project risks and software size.

The description on this chapter provides the answer to RQ-2 (Research Question 2) regarding the impact of identified project risk to the COCOMO effort estimation approach.

Chapter 4

Evaluation by Project Data

The purpose of this chapter is to evaluate the feasibility of using the Fuzzy-ExCOM model to improve the sensitivity level of risk assessment using the Expert COCOMO Model and the ability of the model to provide a contingency allowance for the COCOMO Effort estimation.

This chapter describes the implementation of the Fuzzy-ExCOM model using publicly available COCOMO data points and is organized as follows: Section 4.1 describes the main evaluation steps, and Section 4.2 describes the performance evaluation metrics used in this evaluation. The risk model evaluation is described in Section 4.3 and the evaluation of the contingency allowance model is described in Section 4.4.

4.1 Evaluation Steps

The evaluation activity is basically feeding the model with project data to test the validity, behaviour, and performance of the model. A Fuzzy-ExCOM evaluation is conducted in the following steps: data collection, risk model evaluation, contingency model evaluation, and data analysis as shown in Figure 4.1.

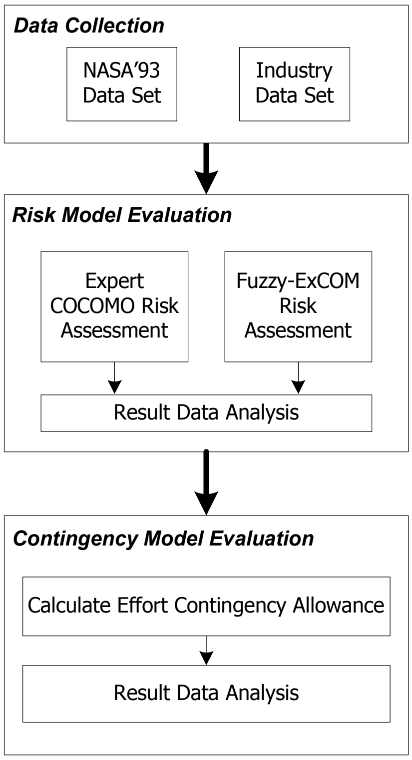


Figure 4.1 Fuzzy-ExCOM Model Evaluation Steps

4.2 Performance Evaluation Metrics

Several metrics are used in this research to evaluate the performance of models and to gain useful insights from calculating the results obtained from the model. The metrics that are used in this thesis are Relative Error (RE), Magnitude Relative Error (MRE), Prediction Level PRED(L), and Correlation Coefficient.

- **Relative Error (RE)**

  The RE is calculated to gain an understanding of the accuracy of the estimates reached by comparing the actual value to the estimated value and expressing the result as a percentage. In the effort estimation stage of project planning, a higher RE suggests that more/less effort than was predicted will be required. The RE formula can be described as follows:

$$RE = \left(\frac{actual - estimated}{actual}\right) \times 100\% \qquad \text{…. (4.1)}$$

Where “actual” is the actual software project effort and “estimated” is the estimated software project effort.

A positive RE value means the actual project effort is higher than estimated effort, and a negative RE value means the actual project effort is smaller than estimated project effort.

- **Magnitude of Relative Error (MRE)**

The MRE is the absolute value of the relative error and can be described according to the following formula:

$$MRE = \frac{|actual - estimated|}{actual} \qquad \text{…. (4.2)}$$

The MRE is used to measure the error contained in the estimated value regardless of whether the error is positive or negative.

- **Prediction Level (PRED)**

The other metric in estimation is the prediction at level L and can be described as follows:

$$PRED\ (L) = \frac{k}{n} \qquad \text{…. (4.3)}$$

Where $\boldsymbol{k}$ is the total number of projects where MRE is less than or equal to $\boldsymbol{L}$ and $\boldsymbol{n}$ is the total number of projects. PRED calculates the ratio of a project’s MRE that falls into the selected range (L) of the total projects. Thus, PRED(25) gives the percentage of projects with an MRE of less than or equal to 25%.

- **Correlation Coefficient**

The correlation coefficient is the statistical metric that is widely used to measure the correlation (linear dependence) between two random variables. This metrics is also known as Pearson's Correlation Coefficient and shown in the following formula:

$$\rho_{X,Y} = \frac{cov\,(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y} \qquad \text{…. (4.4)}$$

Where,

cov (X, Y) is defined as covariance between X and Y.

μ is mean, and σ is standard deviation.

The value of the correlation coefficient is in between -1 to 1. The two parameters can be considered not to have linear relation if their correlation value is equal to 0 (zero). If the correlation value of the two parameters is equal to -1 or 1, it can be said that these two parameters are having a perfect linear relation [82].

4.3 Data Collection

The Fuzzy-ExCOM Model evaluation is tested with 3 data sets. The first data set is NASA'93 public data (93 project data points) [83] and the other data sets are the COCOMO data set from the Turkish Software Industry (12 project data points) [84] and the Industry data set (6 project data points) [85].

Since the original 93 NASA data points and 6 industry data are in the COCOMO’81 format, this thesis uses the 99 data points available in the COCOMO-II format that were used for other research purposes [85]. The complete list of NASA’93 data points is provided in Appendix A, and Appendix B provide the list of Turkish and Industry data set.

4.4 Fuzzy-ExCOM Risk Model Evaluation

The evaluation of Fuzzy-ExCOM Risk Model consists of 3 main processes: make a risk assessment using Expert COCOMO, make a risk assessment using Fuzzy-ExCOM, and calculate coefficient correlation.

The main step in the data analysis is comparing the risk assessment results obtained from the Expert-COCOMO and the Fuzzy-ExCOM methodologies, respectively. Figure 4.2 shows the overall steps in the risk model evaluation.

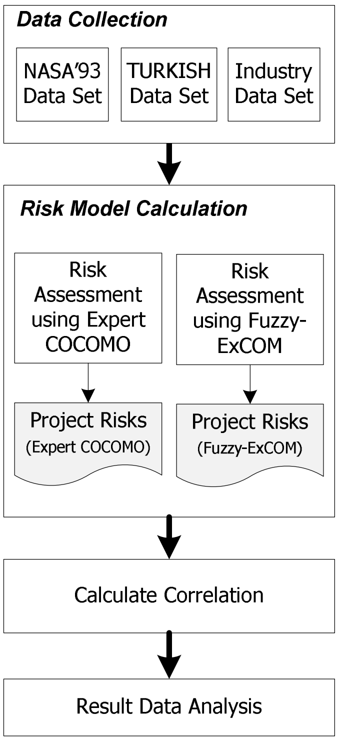


Figure 4.2 Risk Model Evaluation Steps

4.4.1 Expert-COCOMO Risk Assessment

Risk assessment using Expert-COCOMO methodology is based on the application that is posted on the USC website [70]. The Expert-COCOMO risk assessment outputs are: project risk, schedule risk, product risk, platform risk, personnel risk, process risk, and reuse risk.

From the NASA'93 data set of 93 data points, Expert-COCOMO determined that 92 projects were categorized as low risk and only 1 project (project id. 89) was categorized as being a moderate risk project. Table 4.1 provide the partial list of risk assessment results using Expert-COCOMO for NASA'93 data set. The complete list results for NASA'93 data set are listed in Appendix D. The result from TURKISH data set and INDUSTRY data set are shown in Table 4.2 and Table 4.3 respectively.

Table 4.1 Project Risk Assessment using Expert-COCOMO (NASA'93 Data Set)

Project ID	Size (KSLOC)	Actual Effort (person-mo)	Risk Level	Project Risk	Schedule Risk	Product Risk	Platform Risk	Personnel Risk	Process Risk	Reuse Risk
76	162.00	756.00	Low	2.00	2.80	3.00	0.00	0.00	8.30	0.00
77	352.00	1200.00	Low	3.10	2.80	6.40	0.00	0.00	11.80	0.00
78	165.00	97.00	Low	3.10	2.80	6.30	0.00	0.00	11.60	0.00
79	60.00	409.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
80	100.00	703.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
81	32.00	1350.00	Low	0.30	0.00	1.40	0.00	0.00	0.00	0.00
82	53.00	480.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
83	41.00	599.00	Low	0.70	2.70	0.00	0.00	0.90	0.00	0.00
84	24.00	430.00	Low	0.70	2.70	0.00	0.00	0.90	0.00	0.00
85	165.00	4178.20	Low	0.30	0.00	1.50	0.00	0.00	0.00	0.00
86	65.00	1772.50	Low	0.30	0.00	1.50	0.00	0.00	0.00	0.00
87	70.00	1645.90	Low	0.30	0.00	1.50	0.00	0.00	0.00	0.00
88	50.00	1924.50	Low	1.30	0.00	3.60	0.00	0.00	3.90	0.00
89	7.25	648.00	Moderate	7.00	0.00	11.70	7.20	8.50	2.70	0.00
90	233.00	8211.00	Low	0.30	0.00	1.50	0.00	0.00	0.00	0.00
91	16.30	480.00	Low	0.30	0.00	1.40	0.00	0.00	0.00	0.00
92	6.20	12.00	Low	0.30	0.00	1.40	0.00	0.00	0.00	0.00
93	3.00	38.00	Low	0.30	0.00	1.30	0.00	0.00	0.00	0.00

Table 4.2 Project Risk Assessment using Expert-COCOMO (TURKISH Data Set)

Project ID.	Size (KSLOC)	Actual Effort (staff-mo)	Risk Level	Project Risk	Schedule Risk	Personnel Risk	Process Risk	Product Risk	Platform Risk	Reuse Risk
T01	3.00	1.20	Low	0.30	0.00	1.20	0.00	0.00	0.00	0.00
T02	2.00	2.00	Low	0.60	0.00	2.70	0.00	0.00	0.00	0.00
T03	4.25	4.50	Low	0.60	0.00	2.70	0.00	0.00	0.00	0.00
T04	10.00	3.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
T05	15.00	4.00	Low	0.30	0.00	1.20	0.00	0.00	0.00	0.00
T06	40.53	22.00	Low	0.70	2.90	0.00	0.00	0.00	3.00	0.00
T07	4.05	2.00	Low	0.70	2.90	0.00	0.00	0.00	3.00	0.00
T08	31.85	5.00	Low	1.60	0.00	0.00	6.30	2.10	0.00	0.00
T09	114.28	18.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
T10	23.11	4.00	Low	3.50	6.20	1.80	9.50	3.10	0.00	0.00
T11	1.37	1.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
T12	1.61	2.10	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 4.3 Project Risk Assessment using Expert-COCOMO (INDUSTRY Data Set)

Project ID.	Size (KSLOC)	Actual Effort (staff-mo)	Risk Level	Project Risk	Schedule Risk	Personnel Risk	Process Risk	Product Risk	Platform Risk	Reuse Risk
I1	196.60	638.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	1.17
I2	51.80	185.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.83
I3	64.10	332.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	1.50
I4	131.00	619.90	Low	0.00	1.40	0.00	0.00	0.00	0.00	2.03
I5	13.30	64.80	Low	0.00	2.80	0.00	0.00	0.00	0.00	1.50
I6	19.90	76.60	Low	0.00	0.00	0.00	0.00	0.00	0.00	2.44

4.4.2 Fuzzy-ExCOM Risk Assessment

The Fuzzy-ExCOM Risk Model is an improvement over the Expert-COCOMO Model for risk assessment because it implements fuzzy logic in its methodology. The fuzzy logic for the Fuzzy-ExCOM risk model is based on 31 risk rules from Expert-COCOMO. A sample of risk rules is provided in Figure 4.3 and described in more detail in Appendix C. MATLAB R2009b is used as the main tool in the implementation of the Fuzzy-ExCOM risk model. A sample of Fuzzy-ExCOM implementation using MATLAB is provided in Figure 4.4 and a sample input of fuzzification is provided in Figure 4.5.

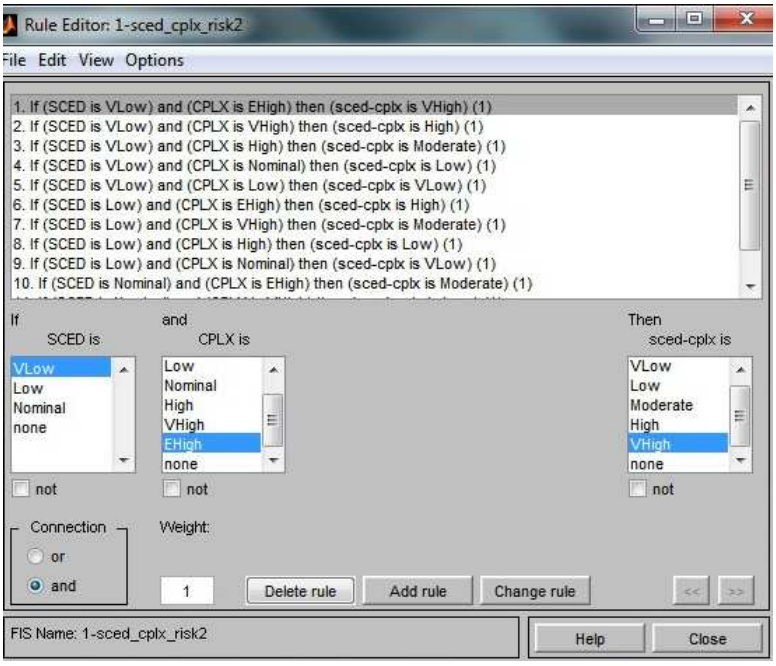


Figure 4.3 SCED Risk Rules Implementation

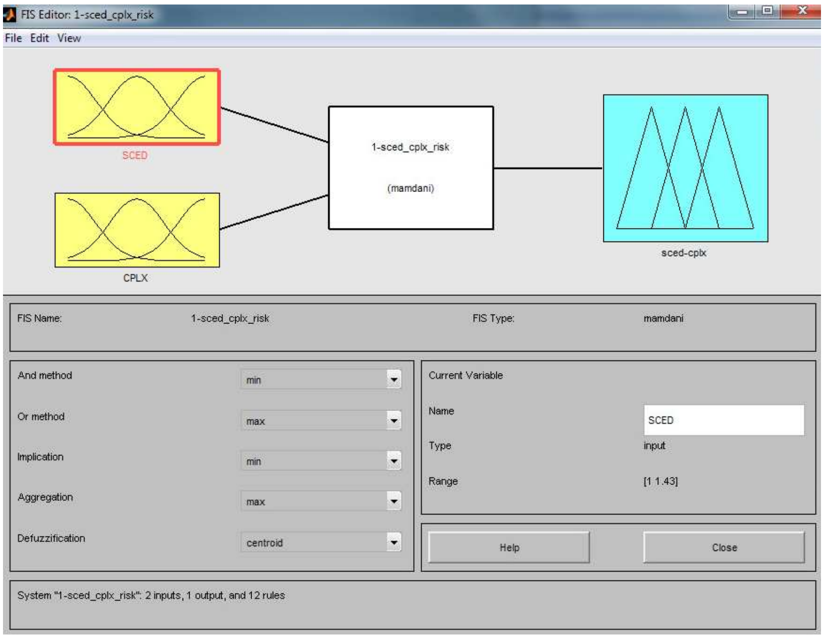


Figure 4.4 SCED_CPLX Implementation

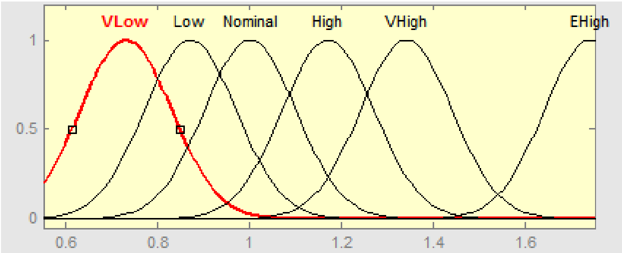


Figure 4.5 CPLX membership function

In the risk assessment results obtained from the NASA’93 data set using the Fuzzy-ExCOM approach, the projects were categorized as: low risk, medium risk, and high risk as follows:

- 21 projects were considered to be Low Risk Projects
- 55 projects were considered to be Moderate Risk Projects
- 17 projects were considered to be High Risk Projects

Table 4.4 show the partial risk assessment results using Fuzzy-ExCOM for NASA’93 data set and the complete results sorted by Risk Level are set out in Appendix E.

The complete Risk Assessment results for using Fuzzy-ExCOM for TURKISH data set and INDUSTRY data set are shown in Table 4.5 and Table 4.6 respectively.

Table 4.4 Project Risk Assessment Results using Fuzzy-ExCOM (NASA'93 Data Set)

Project ID.	Size (KSLOC)	Actual Effort (staff-mo)	Risk Category	Project Risk	Schedule Risk	Personnel Risk	Process Risk	Product Risk	Platform Risk	Reuse Risk
1	25.90	117.60	Moderate	5.19	7.80	9.90	8.34	4.77	6.37	1.54
2	24.60	117.60	Moderate	5.19	7.80	9.90	8.33	4.77	6.37	1.54
3	7.70	31.20	Moderate	5.18	7.80	9.88	8.30	4.76	6.37	1.54
4	8.20	36.00	Moderate	5.18	7.80	9.88	8.31	4.76	6.37	1.54
5	9.70	25.20	Moderate	5.18	7.80	9.88	8.31	4.76	6.37	1.54
6	2.20	8.40	Moderate	5.17	7.79	9.85	8.27	4.74	6.37	1.54
7	3.50	10.80	Moderate	5.17	7.79	9.86	8.28	4.75	6.37	1.54
8	66.60	352.80	Moderate	5.20	7.80	9.92	8.36	4.78	6.37	1.54
9	7.50	72.00	Low	3.77	4.78	8.04	6.46	3.75	3.59	1.54
10	20.00	72.00	Low	4.43	5.29	7.72	9.56	4.55	4.36	1.54
11	6.00	24.00	Low	4.18	4.82	7.57	9.19	4.13	3.96	1.54
12	100.00	360.00	Low	4.27	4.86	7.69	9.56	4.25	3.96	1.54
13	11.30	36.00	Low	4.60	5.20	9.10	9.27	4.15	3.96	2.67
14	100.00	215.00	Moderate	5.42	6.45	11.35	9.94	4.33	4.26	4.14
15	20.00	48.00	Low	4.22	4.83	7.62	9.34	4.18	3.96	1.54
16	100.00	360.00	Moderate	5.27	5.67	11.43	10.02	4.44	4.28	3.50
17	150.00	324.00	Low	4.35	4.86	7.98	9.62	4.27	4.23	1.54
18	31.50	60.00	Low	4.23	4.88	7.63	9.62	4.28	3.62	1.54
19	15.00	48.00	Low	4.07	4.31	7.61	9.30	4.16	3.43	1.54
20	32.50	60.00	Low	4.35	4.88	8.06	9.62	4.28	4.05	1.54
21	19.70	60.00	Moderate	5.19	7.80	9.90	8.33	4.77	6.37	1.54
22	66.60	300.00	Moderate	5.20	7.80	9.92	8.36	4.78	6.37	1.54
23	29.50	120.00	Moderate	5.19	7.80	9.90	8.34	4.77	6.37	1.54
24	15.00	90.00	Low	4.57	5.24	8.77	9.56	4.41	4.16	1.95
25	38.00	210.00	Low	4.88	5.57	9.23	9.99	4.77	4.87	1.95

Project ID.	Size (KSLOC)	Actual Effort (staff-mo)	Risk Category	Project Risk	Schedule Risk	Personnel Risk	Process Risk	Product Risk	Platform Risk	Reuse Risk
76	162.00	756.00	High	17.07	26.73	20.48	47.74	14.59	15.48	2.32
77	352.00	1200.00	High	18.90	27.40	21.10	54.03	20.67	15.48	2.32
78	165.00	97.00	High	18.63	27.12	20.94	52.74	20.40	15.48	2.32
79	60.00	409.00	Moderate	9.76	14.04	19.74	13.20	13.20	10.31	2.32
80	100.00	703.00	Moderate	9.77	14.05	19.75	13.26	13.22	10.31	2.32
81	32.00	1350.00	High	18.52	21.33	41.55	29.18	20.15	19.23	6.71
82	53.00	480.00	Moderate	12.37	19.05	19.13	23.67	13.08	15.07	2.32
83	41.00	599.00	Moderate	14.48	22.79	30.87	21.86	12.11	10.34	10.03
84	24.00	430.00	Moderate	14.34	22.65	30.75	21.28	11.90	10.34	10.03
85	165.00	4178.20	Moderate	14.27	18.13	22.68	28.15	20.33	14.85	2.32
86	65.00	1772.50	Moderate	14.32	18.17	22.71	28.38	20.43	14.85	2.32
87	70.00	1645.90	Moderate	14.43	18.25	22.76	28.86	20.61	14.85	2.32
88	50.00	1924.50	High	16.27	18.69	26.85	30.99	27.68	14.85	2.32
89	7.25	648.00	High	23.74	25.93	54.30	38.33	29.78	20.23	8.52
90	233.00	8211.00	Moderate	14.26	18.12	22.67	28.09	20.31	14.85	2.32
91	16.30	480.00	High	17.24	21.50	36.81	28.75	20.04	14.81	6.71
92	6.20	12.00	High	17.12	21.41	36.67	28.21	19.91	14.81	6.71
93	3.00	38.00	High	17.03	21.34	36.57	27.82	19.81	14.81	6.71

Table 4.5 Project Risk Assessment Results using Fuzzy-ExCOM (TURKISH Data Set)

Project ID.	Size (KSLOC)	Actual Effort (staff-mo)	Risk Level	Project Risk	Schedule Risk	Personnel Risk	Process Risk	Product Risk	Platform Risk	Reuse Risk
T01	3.00	1.20	Low	3.68	5.21	6.06	8.61	3.95	2.82	0.78
T02	2.00	2.00	Low	4.01	5.24	6.20	9.60	5.42	2.91	0.54
T03	4.25	4.50	Low	4.16	5.32	6.34	10.34	5.53	2.97	0.54
T04	10.00	3.00	Low	3.94	5.51	6.67	7.56	4.47	4.26	0.90
T05	15.00	4.00	Low	4.58	7.04	8.60	7.87	4.65	4.05	1.95
T06	40.53	22.00	Low	4.70	7.66	6.64	10.01	2.67	6.58	1.51
T07	4.05	2.00	Low	4.94	7.88	6.56	12.05	2.50	6.58	1.25
T08	31.85	5.00	Low	4.79	4.78	11.22	10.94	3.13	4.19	1.50
T09	114.28	18.00	Moderate	5.18	5.91	10.94	12.21	2.78	6.14	0.66
T10	23.11	4.00	Moderate	5.10	7.63	10.75	12.65	2.56	3.80	0.66
T11	1.37	1.00	Low	3.38	4.87	6.88	5.86	3.14	2.92	1.54
T12	1.61	2.10	Low	3.95	5.05	9.04	5.67	3.68	5.16	0.83

Table 4.6 Project Risk Assessment Results using Fuzzy-ExCOM (INDUSTRY Data Set)

Project ID.	Size (KSLOC)	Actual Effort (staff-mo)	Risk Level	Project Risk	Schedule Risk	Personnel Risk	Process Risk	Product Risk	Platform Risk	Reuse Risk
I1	196.60	638.00	Low	4.49	7.25	7.78	9.20	4.48	3.65	1.17
I2	51.80	185.00	Low	4.40	7.60	6.81	8.83	4.42	4.31	0.83
I3	64.10	332.00	Low	4.58	5.48	9.71	8.20	4.46	4.83	1.50
I4	131.00	619.90	Moderate	5.24	5.80	10.67	10.43	4.63	5.56	2.03
I5	13.30	64.80	Moderate	6.32	6.40	11.16	16.04	6.49	5.56	1.50
I6	19.90	76.60	Low	4.97	5.95	11.21	8.23	4.41	4.80	2.44

4.4.3 Calculate Correlation Coefficient

The Correlation Coefficient is calculated to explain the degree of correlation between project risks and other project parameters. It also provides the information about the sensitivity of project risks to the variations in these parameters. This research calculates the correlation coefficient between the project risk with the software size and also with the actual project effort.

Software size in a software development project is having a proportional relationship with project risk; the larger software size means a higher project risk [59]. Project risks also have a relationship with project effort because the problems in project execution that come from the potential project risks will be carried over to project effort [19].

Table 4.7 shows the correlation between project risk versus software size and actual effort based on Expert-COCOMO and the Fuzzy-ExCOM approach for the NASA'93 data set. The correlation chart diagram for risk against software size for NASA'93 data set is shown in Figure 4.6.

Table 4.7 Risk Correlation with Size and Actual Effort (NASA'93 Data Set)

corr *(NASA93 data set)*	***Size (KSLOC)***	***ACT Effort (staff-mo)***
Expert COM Risk	0.05	0.02
fuzzy-ExCOM Risk	0.25	0.31

Table 4.8 shows the correlation results for the TURKISH data set and the correlation results for the INDUSTRY data set is shown in Table 4.9.

Table 4.8 Risk Correlation with Size and Actual Effort (TURKISH Data Set)

corr *(TURKISH data set)*	***Size (KSLOC)***	***ACT Effort (staff-mo)***
Expert COM Risk	0.00	-0.04
fuzzy-ExCOM Risk	0.63	0.53

Table 4.9 Risk Correlation with Size and Actual Effort (INDUSTRY Data Set)

corr *(INDUSTRY data set)*	***Size (KSLOC)***	***ACT Effort (staff-mo)***
Expert COM Risk	0.00	0.00
fuzzy-ExCOM Risk	-0.42	-0.37

.

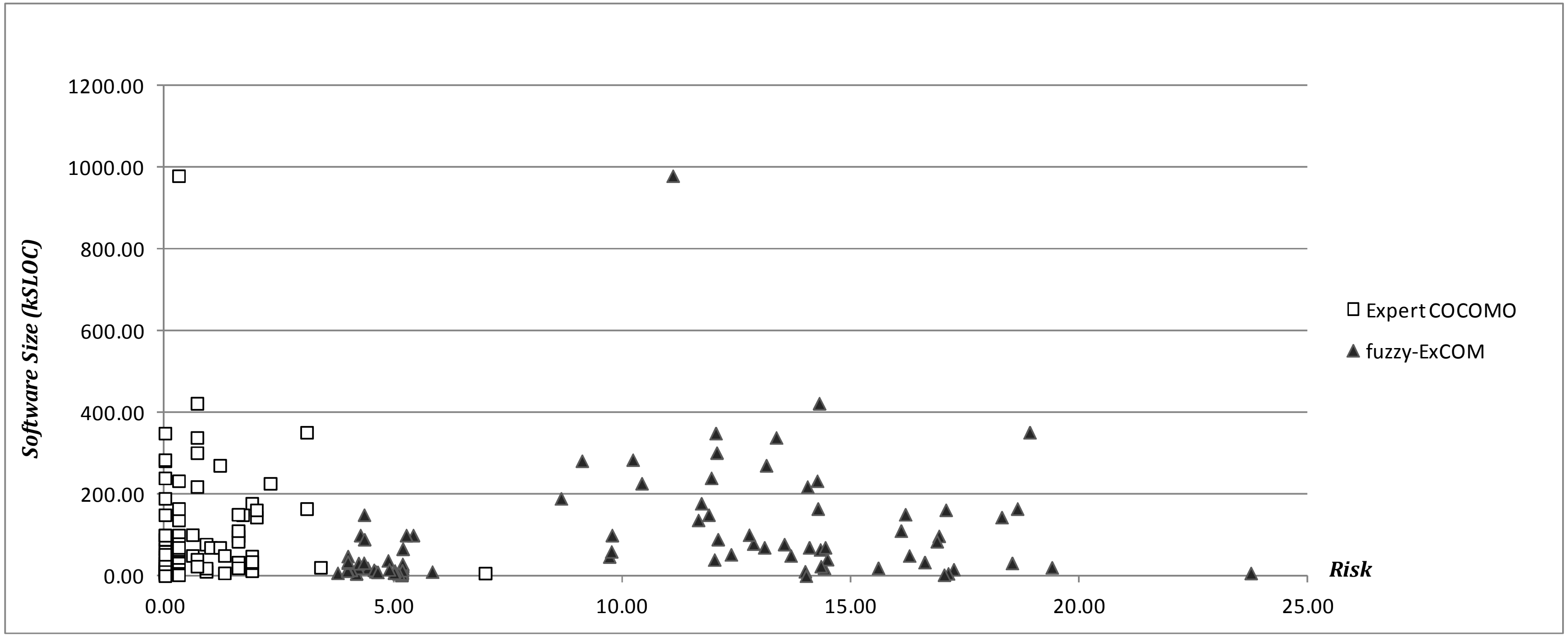


Figure 4.6 Project Risk Correlation with Software Size (NASA'93 Data Set)

4.4.4 Analysis on Risk Model Evaluation Results

In the NASA'93 data set, from a total of 93 data points, Expert-COCOMO categorized 92 projects as being low risk and only 1 (project id.89) as medium risk. For the 92 projects categorized as being low risk, the project risk value and the module risks value were mostly zero as shown in Table 4.1 and Appendix D. The risk assessment using Expert-COCOMO to TURKISH and INDUSTRY data set provides similar results as shown in Table 4.2 and Table 4.3.

If the risk assessment using Expert-COCOMO were to be conducted during a project planning session, the project manager could conclude that the project would go smoothly and that its completion would be achieved using the estimated effort with few errors because the project risk was low.

On the other hand, the risk calculation result using Fuzzy-ExCOM to NASA'93, TURKISH, and INDUSTRY data sets categorizes the projects as low, moderate, and high risk projects as shown in Table 4.4, Table 4.5, Table 4.6, and Appendix E,.

For the NASA'93 data set, there were 21 projects, which were considered to be low risk projects, 55 projects, which were considered to be moderate risk projects, and 17 projects, which were considered to be high risk projects.

For the TURKISH data set, there were 10 projects, which were considered to be low risk projects, and 2 projects, which were considered to be moderate risk projects. For the INDUSTRY data set, there were 4 projects, which were considered to be low risk projects, and 2 projects, which were considered to be moderate risk projects.

The correlation calculation for the two risk assessment approaches to 3 data sets as described at Table 4.7, Table 4.8, and Table 4.9 show that Fuzzy-ExCOM risk assessment results are producing a higher correlation with software size and actual effort for all of the data sets compared to the results from Expert-COCOMO.

Thus, it can be said that Fuzzy-ExCOM provides a better and more sensitive risk assessment result compared to the original method, and thus provides more valuable information to the project manager for planning purposes.

4.5 Fuzzy-ExCOM Effort Contingency Model Evaluation

The Fuzzy-ExCOM Effort Contingency Model calculates an effort contingency allowance to accommodate the risks and assumptions that are used in effort estimations. The model evaluation consists of the following steps: first, estimate effort and calculate error estimate (RE/MRE) , the second step is to calculate the effort contingency value based on the project risk and project size, and then calculate the maximum and minimum estimated effort value based on a contingency allowance. The last step is to conduct an analysis of the evaluation model itself. The overall effort contingency model evaluation steps are shown in Figure 4.7.

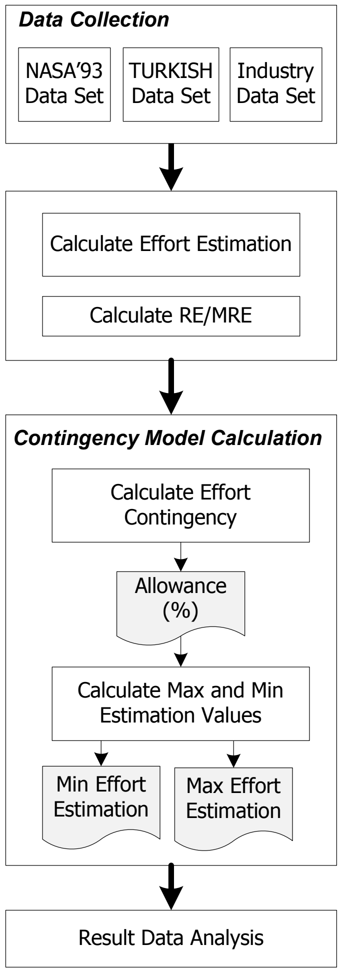


Figure 4.7 Contingency Model Evaluation Step

4.5.1 Estimate Effort using COCOMO II

Effort estimation is calculated to provide the base effort estimate value for each data set and the benchmark for the contingency value. The effort estimation value for the NASA'93 data set has been collected from a previous research article [85] while the effort estimation value for the TURKISH and the INDUSTRY data sets has been calculated using an online COCOMO-II application [86].

This effort estimation value is required in order to calculate Relative Error and Magnitude Relative Error (RE/MRE), which provides an indication of the estimation accuracy as compared to the actual effort estimation value.

4.5.2 Effort Contingency Calculation

The Fuzzy-ExCOM Effort Contingency Model is based on fuzzy techniques which is involved the fuzzification process, the inference rule, and the defuzzification process. Figure 4.8 shows the implementation of the model using Matlab. The input fuzzification and output defuzzification steps are shown in Figure 4.9 and Figure 4.10, respectively. Figure 4.11 shows the implementation of the contingency rules based on Table 3.2 in Chapter 3, which is part of the inference process.

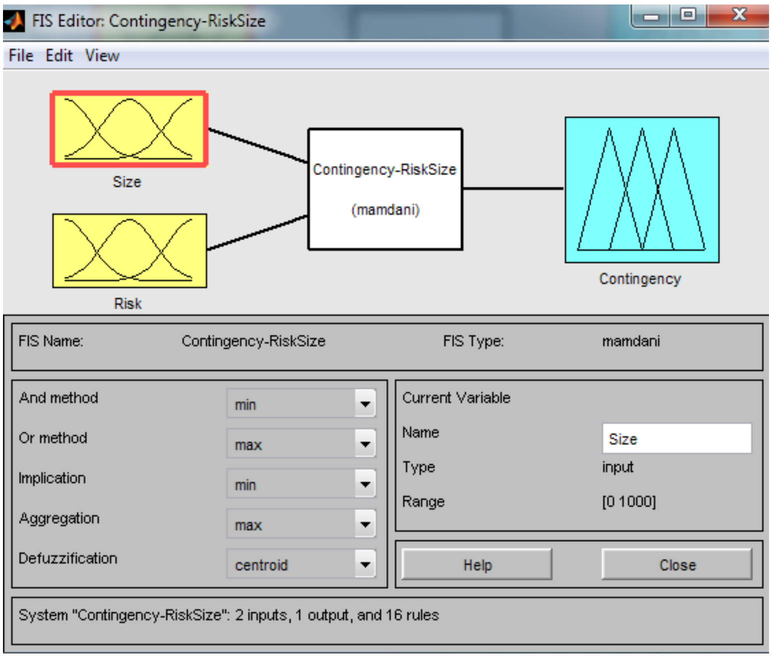


Figure 4.8 Contingency Implementation

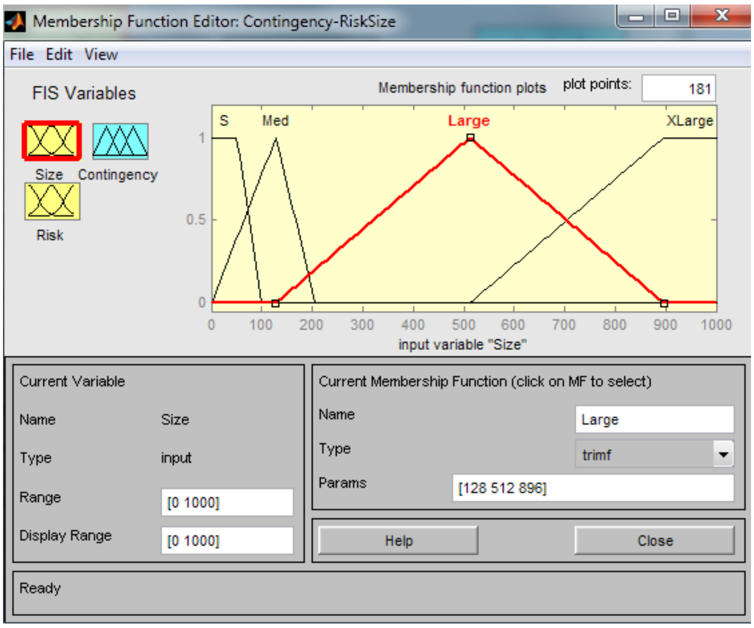


Figure 4.9a Size Input Fuzzification

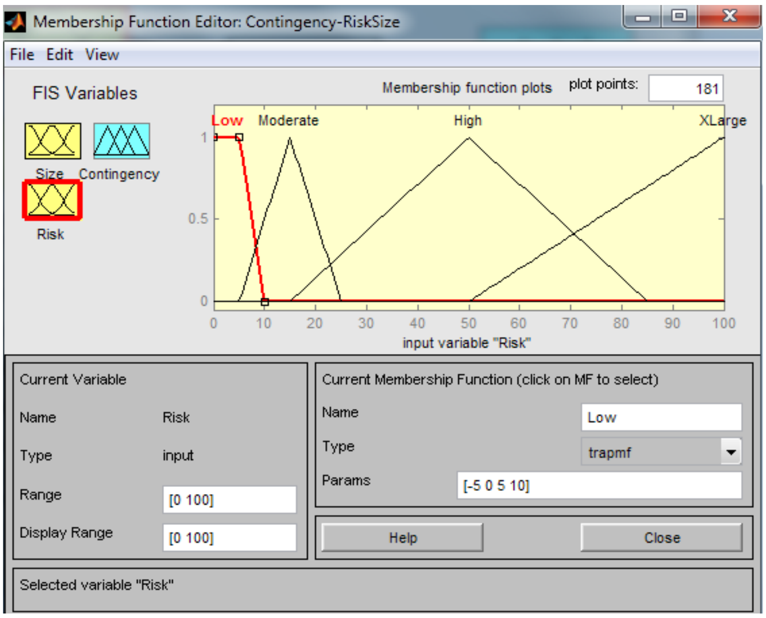


Figure 4.9b Risk Input Fuzzification

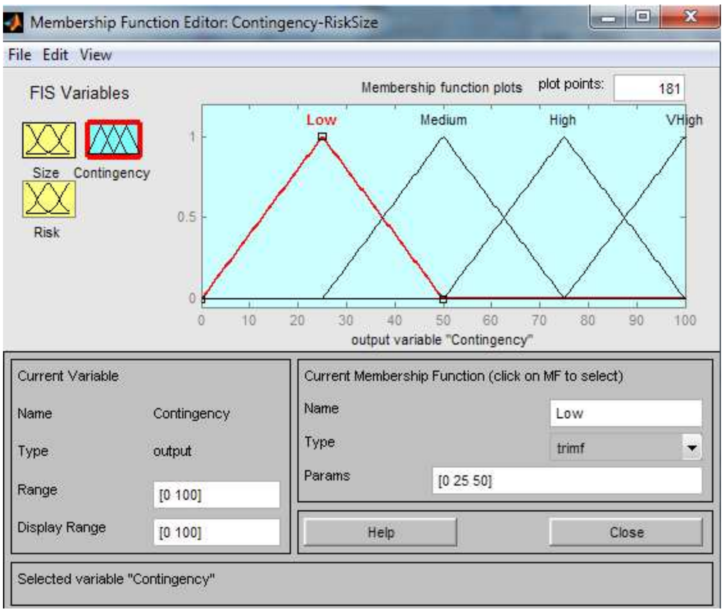


Figure 4.10 Contingency Output Defuzzification

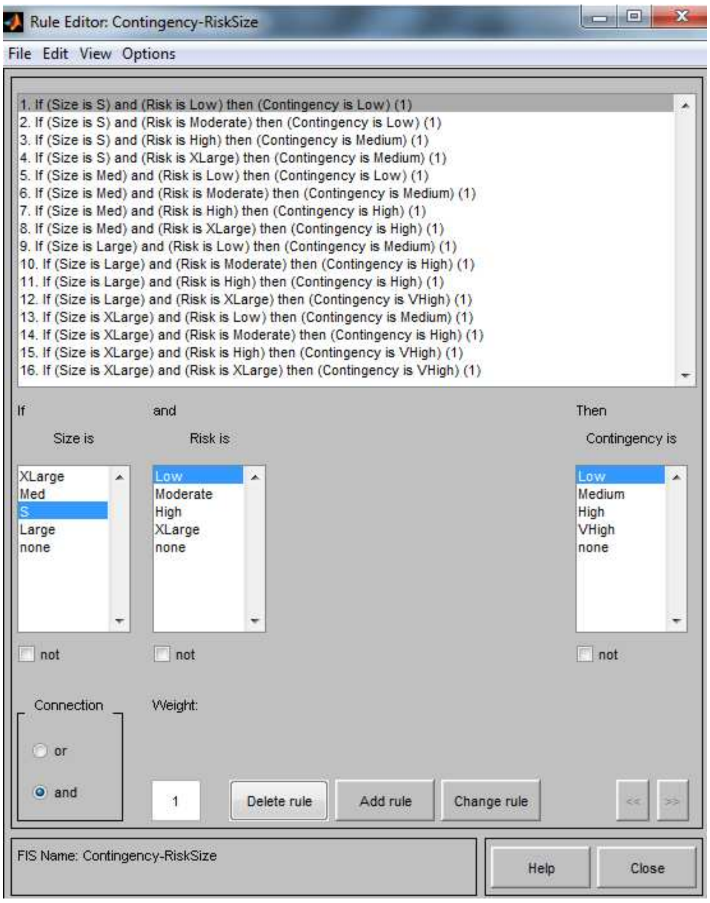


Figure 4.11 Contingency Rule Implementation

The contingency model calculation provides the contingency allowance value based on the level of project risk and software size. The output of this calculation is a contingency allowance that can be used to calculate the maximum and the minimum estimation value. Table 4.10 shows the partial results (50 data points) for NASA’93 data set. The overall results for NASA’93 data set is described in more detail in Appendix F. The results from INDUSTRY and TURKISH data set are shown in Table 4.11.

Table 4.10 Partial Results of Effort Contingency (NASA'93 Data Set)

Proj. ID.	Size (KSLOC)	Size Category	Project Risk	Risk Category	Actual Effort (staff-mo)	COCOMO II	RE	MRE	Contingency Allowance	Contingency Category	MIN Effort Est Value	MAX Effort Est Value
1	25.90	Small	5.19	Moderate	117.60	104.97	11%	11%	25.7%	Medium	77.99	131.95
2	24.60	Small	5.19	Moderate	117.60	99.49	15%	15%	25.7%	Medium	73.92	125.06
3	7.70	Small	5.18	Moderate	31.20	29.69	5%	5%	25.7%	Medium	22.06	37.32
4	8.20	Small	5.18	Moderate	36.00	31.70	12%	12%	25.7%	Medium	23.55	39.85
5	9.70	Small	5.18	Moderate	25.20	37.75	-50%	50%	25.7%	Medium	28.05	47.45
6	2.20	Small	5.17	Moderate	8.40	8.06	4%	4%	25.6%	Medium	6.00	10.12
7	3.50	Small	5.17	Moderate	10.80	13.06	-21%	21%	25.6%	Medium	9.72	16.40
8	66.60	Medium	5.20	Moderate	352.80	280.63	20%	20%	25.8%	Medium	208.23	353.03
9	7.50	Small	3.77	Low	72.00	24.82	66%	66%	25.0%	Low	18.62	31.03
10	20.00	Small	4.43	Low	72.00	36.80	49%	49%	25.0%	Low	27.60	46.00
11	6.00	Small	4.18	Low	24.00	11.04	54%	54%	25.0%	Low	8.28	13.80
12	100.00	Medium	4.27	Low	360.00	201.60	44%	44%	25.0%	Low	151.20	252.00
13	11.30	Small	4.60	Low	36.00	28.36	21%	21%	25.0%	Low	21.27	35.45
14	100.00	Medium	5.42	Moderate	215.00	479.18	-123%	123%	26.5%	Medium	352.20	606.16
15	20.00	Small	4.22	Low	48.00	39.40	18%	18%	25.0%	Low	29.55	49.25
16	100.00	Medium	5.27	Moderate	360.00	411.53	-14%	14%	26.0%	Medium	304.53	518.53
17	150.00	Large	4.35	Low	324.00	451.79	-39%	39%	30.1%	Medium	315.80	587.78
18	31.50	Small	4.23	Low	60.00	73.72	-23%	23%	25.0%	Low	55.29	92.15
19	15.00	Small	4.07	Low	48.00	29.07	39%	39%	25.0%	Low	21.80	36.34
20	32.50	Small	4.35	Low	60.00	122.25	-104%	104%	25.0%	Low	91.69	152.81
21	19.70	Small	5.19	Moderate	60.00	78.95	-32%	32%	25.7%	Medium	58.66	99.24
22	66.60	Medium	5.20	Moderate	300.00	280.63	6%	6%	26.0%	Medium	207.67	353.59
23	29.50	Small	5.19	Moderate	120.00	120.20	0%	0%	25.7%	Medium	89.31	151.09
24	15.00	Small	4.57	Low	90.00	57.99	36%	36%	25.0%	Low	43.49	72.49
25	38.00	Small	4.88	Low	210.00	163.26	22%	22%	25.0%	Low	122.45	204.08
26	10.00	Small	4.65	Low	48.00	30.94	36%	36%	25.0%	Low	23.21	38.68
27	15.40	Small	3.99	Low	70.00	66.08	6%	6%	25.0%	Low	49.56	82.60
28	48.50	Small	4.00	Low	239.00	218.17	9%	9%	25.0%	Low	163.63	272.71
29	16.30	Small	3.99	Low	82.00	70.10	15%	15%	25.0%	Low	52.58	87.63
30	12.80	Small	3.99	Low	62.00	54.50	12%	12%	25.0%	Low	40.88	68.13
31	32.60	Small	4.00	Low	170.00	144.27	15%	15%	25.0%	Low	108.20	180.34
32	35.50	Small	4.00	Low	192.00	157.65	18%	18%	25.0%	Low	118.24	197.06
33	5.50	Small	5.17	Moderate	18.00	20.91	-16%	16%	25.6%	Medium	15.56	26.26
34	10.40	Small	5.84	Moderate	50.00	40.60	19%	19%	27.9%	Medium	29.27	51.93
35	14.00	Small	5.18	Moderate	60.00	55.32	8%	8%	25.7%	Medium	41.10	69.54
36	6.50	Small	5.10	Moderate	42.00	31.54	25%	25%	25.4%	Medium	23.53	39.55
37	13.00	Small	5.02	Moderate	60.00	59.66	1%	1%	25.1%	Medium	44.69	74.63
38	90.00	Medium	4.36	Low	444.00	346.90	22%	22%	25.0%	Low	260.18	433.63
39	8.00	Small	5.01	Moderate	42.00	35.71	15%	15%	25.0%	Low	26.78	44.64
40	16.00	Small	4.90	Low	114.00	82.47	28%	28%	25.0%	Low	61.85	103.09
41	177.90	Large	11.73	Moderate	1248.00	1035.91	17%	17%	61.6%	High	397.79	1674.03
42	302.00	Large	12.06	Moderate	2400.00	1120.94	53%	53%	75.0%	High	280.24	1961.65
43	282.10	Large	9.12	Moderate	1368.00	830.26	39%	39%	67.5%	High	269.83	1390.69
44	284.70	Large	10.23	Moderate	973.00	994.21	-2%	2%	75.0%	High	248.55	1739.87
45	79.00	Medium	12.86	Moderate	400.00	272.93	32%	32%	40.1%	Medium	163.49	382.37
46	423.00	Large	14.30	Moderate	2400.00	904.51	62%	62%	75.0%	High	226.13	1582.89
47	190.00	Large	8.66	Moderate	420.00	382.38	9%	9%	55.8%	High	169.01	595.75
48	47.50	Small	9.71	Moderate	252.00	157.89	37%	37%	37.5%	Medium	98.68	217.10
49	21.00	Small	19.39	High	107.00	152.63	-43%	43%	38.1%	Medium	94.48	210.78
50	78.00	Medium	13.54	Moderate	571.40	339.63	41%	41%	39.9%	Medium	204.12	475.14

Table 4.11 Results of Effort Contingency (INDUSTRY and TURKISH Data Set)

Proj. ID.	Size (KSLOC)	Size Category	Project Risk	Risk Category	Actual Effort (staff-mo)	COCOMO II	RE	MRE	Contingency Allowance	Contingency Category	MIN Effort Est Value	MAX Effort Est Value
I01	196.60	Large	4.49	Low	638.00	722.70	-13%	13%	40.3%	Medium	431.45	1013.95
I02	51.80	Medium	4.40	Low	185.00	140.00	24%	24%	25.0%	Low	105.00	175.00
I03	64.10	Medium	4.58	Low	332.00	256.70	23%	23%	25.0%	Low	192.53	320.88
I04	131.00	Large	5.24	Moderate	619.90	745.20	-20%	20%	26.3%	Medium	549.21	941.19
I05	13.30	Small	6.32	Moderate	64.80	68.90	-6%	6%	28.6%	Medium	49.19	88.61
I06	19.90	Small	4.97	Low	76.60	92.70	-21%	21%	25.0%	Low	69.53	115.88
T01	3.00	Small	3.68	Low	1.20	3.60	-200%	200%	25.0%	Low	2.70	4.50
T02	2.00	Small	4.01	Low	2.00	2.90	-45%	45%	25.0%	Low	2.18	3.63
T03	4.25	Small	4.16	Low	4.50	9.30	-107%	107%	25.0%	Low	6.98	11.63
T04	10.00	Small	3.94	Low	3.00	36.20	-1107%	1107%	25.0%	Low	27.15	45.25
T05	15.00	Small	4.58	Low	4.00	63.20	-1480%	1480%	25.0%	Low	47.40	79.00
T06	40.53	Small	4.70	Low	22.00	28.60	-30%	30%	25.0%	Low	21.45	35.75
T07	40.50	Small	4.94	Low	2.00	2.30	-15%	15%	25.0%	Low	1.73	2.88
T08	31.85	Small	4.79	Low	5.00	147.10	-2842%	2842%	25.0%	Low	110.33	183.88
T09	114.28	Medium	5.18	Moderate	18.00	294.00	-1533%	1533%	25.7%	Medium	218.44	369.56
T10	23.11	Small	5.10	Moderate	4.00	63.20	-1480%	1480%	25.4%	Medium	47.15	79.25
T11	1.37	Small	3.38	Low	1.00	0.90	10%	10%	25.0%	Low	0.68	1.13
T12	1.61	Small	3.95	Low	2.10	2.00	5%	5%	25.0%	Low	1.50	2.50

4.5.3 Analysis on Effort Contingency Model Evaluation Result

The Fuzzy-ExCOM Effort Contingency Model for NASA'93 data set provides the contingency allowance value, which is in the range of 25% to 75% of the effort estimation value. The composition of the effort contingency allowance for the NASA'93 data set is described as follows:

- 22 projects (24%) are having a LOW Allowance (0% to 25%)
- 46 projects (49%) are having a MEDIUM Allowance (25.1% to 50%)
- 25 projects (27%) are having a HIGH Allowance (50.1% to 75%)

In the INDUSTRY and TURKISH data set, the contingency allowances are in the range of 25% to 40% with the composition as follows.

- 13 projects (72%) are having a LOW Allowance (0% to 25%)
- 5 projects (28%) are having a MEDIUM Allowance (25.1% to 50%)

The Contingency Allowance Model applied to the NASA’93 data set is shown in Figure 4.12, which describes four effort values: the MIN (minimum estimation value), the MAX (maximum estimation value), the ACTUAL value, and the EST (estimation value). The allowance value is the value that lies between the MIN and the MAX values, which represent the upper and lower levels of the estimation value. The results of Contingency Allowance Model applied to INDUSTRY and TURKISH data set is shown in Figure 4.13.

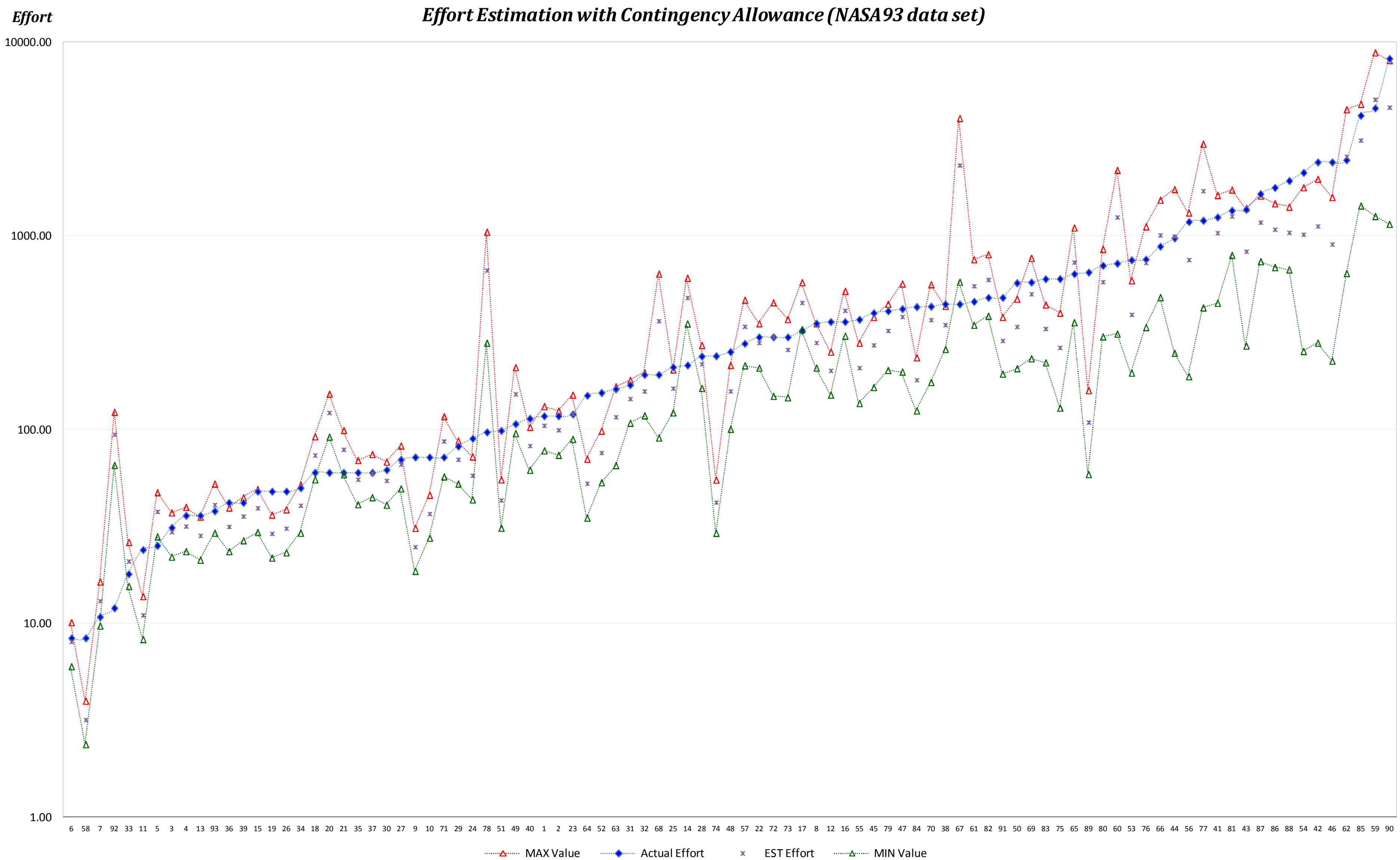


Figure 4.12 Effort Estimation with Contingency Allowance (NASA’93 Data Set)

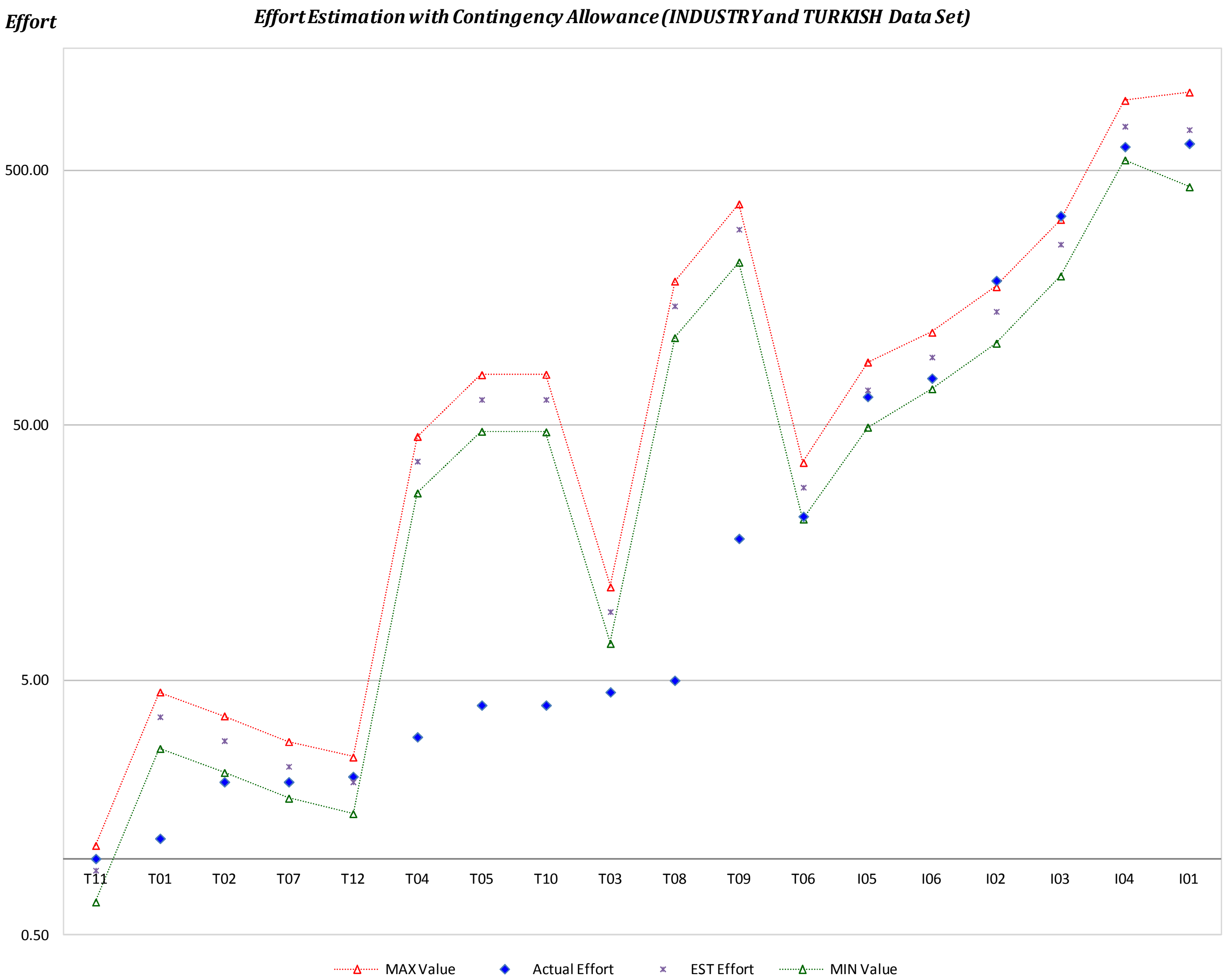


Figure 4.13 Effort Estimation with Contingency Allowance (INDUSTRY and TURKISH Data Set)

From these four types of effort value, the most important value for project planning purposes is the MAX value, which represents the estimate value with additional resources (contingency value) that should be gathered by the project manager to compensate the project risks. The project with ACTUAL value, which is lower than the MAX value, is preferable because the actual amount of resources consumed will be lower than the amount of budgeted resources (estimate value + contingency value).

In the Contingency Allowance Model performance evaluation, the MAX value becomes the main reference point for the performance calculation. The Contingency Allowance Model compares the MAX value to the actual effort value and uses 5 parameters to describe the model performance as follows:

- FIT describes the projects with an ACTUAL value that is lower than MAX value.
- FIT+PRED(25) describes FIT projects AND the projects with an MRE between the ACTUAL and the MAX is less than 25%.
- FIT+PRED(50) describes the FIT projects AND the projects with an MRE between the ACTUAL and the MAX is less than 50%.
- FIT+PRED(75) describes the FIT projects AND the projects with an MRE between the ACTUAL and the MAX is less than 75%.
- FIT+PRED(90) describes the FIT projects AND the projects with an MRE between the ACTUAL and the MAX is less than 90%.

For the performance evaluation purposes, FIT project in the original COCOMO-II Estimation results describes as the project with an ACTUAL value, which is lower than the EST value.

Figure 4.14 shows the graphic of the FIT projects for NASA’93 data set, which have been sorted based on the actual effort and the list of FIT projects is described in Table 4.12.

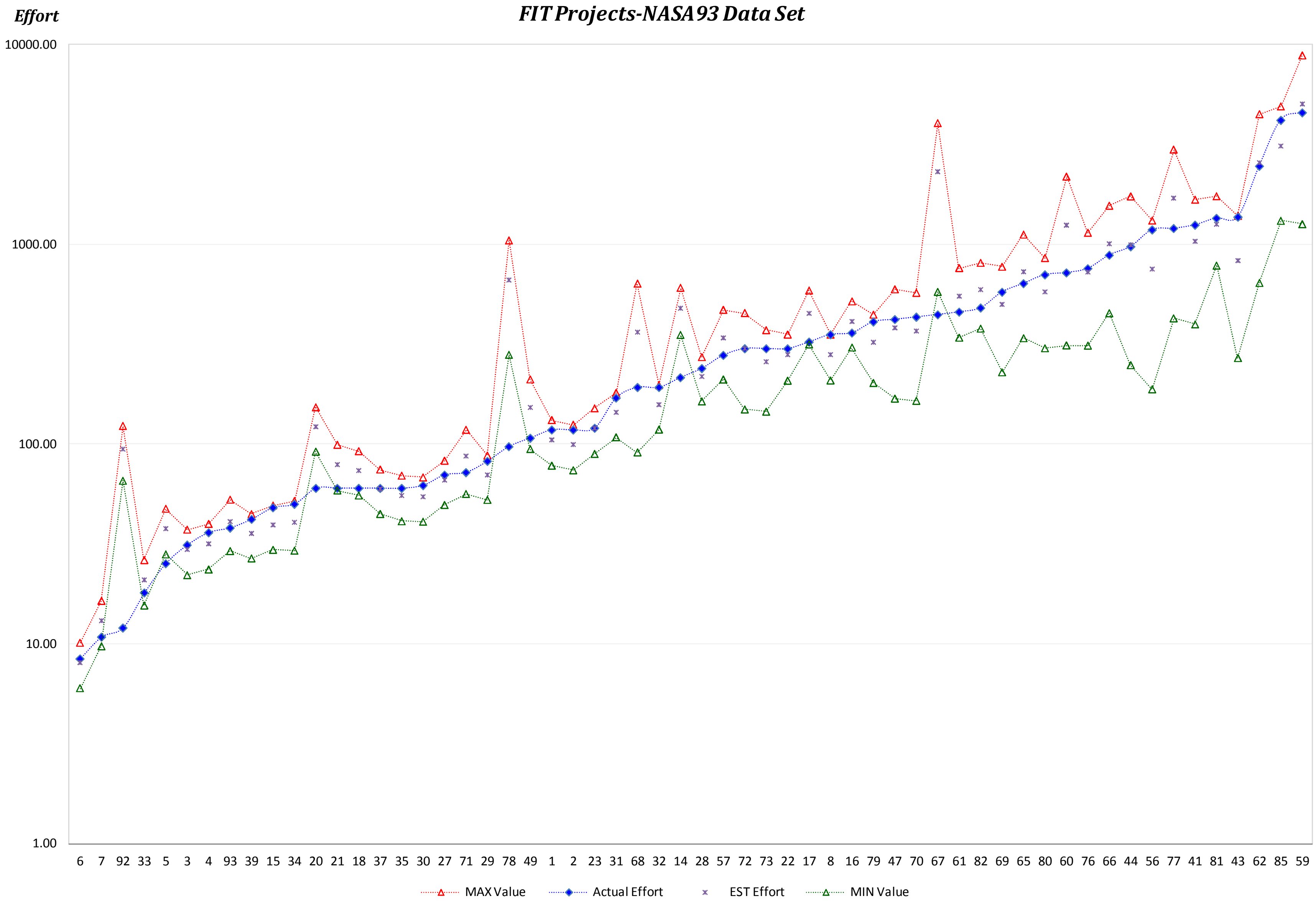


Figure 4.14 FIT Projects (NASA'93 Data Set)

Table 4.12 FIT Project List (NASA'93 Data Set)

Project ID.	Size (KSLOC)	Size Category	Project Risk	Risk Category	COCOMO II	Contingency Allowance	Contingency Category	MIN Eff Est Value	MAX Eff Est Value	Actual Effort (staff-mo)
1	25.90	Small	5.19	Moderate	104.97	26%	Medium	77.99	131.95	117.60
2	24.60	Small	5.19	Moderate	99.49	26%	Medium	73.92	125.06	117.60
3	7.70	Small	5.18	Moderate	29.69	26%	Medium	22.06	37.32	31.20
4	8.20	Small	5.18	Moderate	31.70	26%	Medium	23.55	39.85	36.00
5	9.70	Small	5.18	Moderate	37.75	26%	Medium	28.05	47.45	25.20
6	2.20	Small	5.17	Moderate	8.06	26%	Medium	6.00	10.12	8.40
7	3.50	Small	5.17	Moderate	13.06	26%	Medium	9.72	16.40	10.80
8	66.60	Medium	5.20	Moderate	280.63	26%	Medium	208.23	353.03	352.80
14	100.00	Medium	5.42	Moderate	479.18	27%	Medium	352.20	606.16	215.00
15	20.00	Small	4.22	Low	39.40	25%	Low	29.55	49.25	48.00
16	100.00	Medium	5.27	Moderate	411.53	26%	Medium	304.53	518.53	360.00
17	150.00	Large	4.35	Low	451.79	30%	Medium	315.80	587.78	324.00
18	31.50	Small	4.23	Low	73.72	25%	Low	55.29	92.15	60.00
20	32.50	Small	4.35	Low	122.25	25%	Low	91.69	152.81	60.00
21	19.70	Small	5.19	Moderate	78.95	26%	Medium	58.66	99.24	60.00
22	66.60	Medium	5.20	Moderate	280.63	26%	Medium	207.67	353.59	300.00
23	29.50	Small	5.19	Moderate	120.20	26%	Medium	89.31	151.09	120.00
27	15.40	Small	3.99	Low	66.08	25%	Low	49.56	82.60	70.00
28	48.50	Small	4.00	Low	218.17	25%	Low	163.63	272.71	239.00
29	16.30	Small	3.99	Low	70.10	25%	Low	52.58	87.63	82.00
30	12.80	Small	3.99	Low	54.50	25%	Low	40.88	68.13	62.00
31	32.60	Small	4.00	Low	144.27	25%	Low	108.20	180.34	170.00
32	35.50	Small	4.00	Low	157.65	25%	Low	118.24	197.06	192.00
33	5.50	Small	5.17	Moderate	20.91	26%	Medium	15.56	26.26	18.00
34	10.40	Small	5.84	Moderate	40.60	28%	Medium	29.27	51.93	50.00
35	14.00	Small	5.18	Moderate	55.32	26%	Medium	41.10	69.54	60.00
37	13.00	Small	5.02	Moderate	59.66	25%	Low	44.69	74.63	60.00
39	8.00	Small	5.01	Moderate	35.71	25%	Low	26.78	44.64	42.00
41	177.90	Large	11.73	Moderate	1035.91	62%	High	397.7894	1674.0306	1248.00
43	282.10	Large	9.12	Moderate	830.26	68%	High	269.83	1390.69	1368.00
44	284.70	Large	10.23	Moderate	994.21	75%	High	248.55	1739.87	973.00
47	190.00	Large	8.66	Moderate	382.38	56%	High	169.01	595.75	420.00
49	21.00	Small	19.39	High	152.63	38%	Medium	94.48	210.78	107.00
56	227.00	Large	10.42	Moderate	752.33	75%	High	188.08	1316.58	1181.00
57	70.00	Medium	14.08	Moderate	340.43	38%	Medium	210.73	470.13	278.00
59	980.00	X Large	11.10	Moderate	5048.36	75%	High	1262.09	8834.63	4560.00
60	350.00	Large	12.04	Moderate	1248.08	75%	High	312.02	2184.14	720.00
61	70.00	Medium	13.10	Moderate	550.87	38%	Medium	340.99	760.75	458.00
62	271.00	Large	13.14	Moderate	2564.75	75%	High	641.19	4488.31	2460.00
65	137.00	Large	11.66	Moderate	729.48	54%	High	339.21	1119.75	636.00
66	150.00	Large	11.89	Moderate	1007.39	55%	High	452.32	1562.46	882.00
67	339.00	Large	13.36	Moderate	2312.83	75%	High	578.21	4047.45	444.00
68	240.00	Large	11.94	Moderate	363.67	75%	High	90.92	636.42	192.00
69	144.00	Large	18.29	High	500.66	54%	High	228.80	772.52	576.00
70	151.00	Large	16.18	High	368.39	55%	High	164.67	572.11	432.00
71	34.00	Small	16.61	High	87.12	35%	Medium	56.28	117.96	72.00
72	98.00	Medium	16.92	High	301.24	51%	High	149.11	453.37	300.00
73	85.00	Medium	16.87	High	258.60	44%	Medium	145.33	371.87	300.00
76	162.00	Large	17.07	High	727.12	57%	High	311.21	1143.03	756.00
77	352.00	Large	18.90	High	1704.70	75%	High	426.175	2983.225	1200.00
78	165.00	Large	18.63	High	663.53	58%	High	280.01	1047.05	97.00
79	60.00	Medium	9.76	Moderate	323.70	38%	Medium	202.31	445.09	409.00
80	100.00	Medium	9.77	Moderate	578.40	48%	Medium	301.92	854.88	703.00
81	32.00	Small	18.52	High	1262.18	38%	Medium	782.55	1741.81	1350.00
82	53.00	Medium	12.37	Moderate	593.55	36%	Medium	378.68	808.42	480.00
85	165.00	Large	14.27	Moderate	3109.49	58%	High	1312.20	4906.78	4178.20
92	6.20	Small	17.12	High	94.43	31%	Medium	65.53	123.33	12.00
93	3.00	Small	17.03	High	40.92	29%	Medium	29.14	52.70	38.00

The performance of Contingency Allowance Model with 5 estimation parameters compared to the performance of COCOMO Model is described in Table 4.13.

Table 4.13 Contingency Allowance Model and COCOMO Model Performance Comparison (NASA'93 Data Set)

Total Project = 93	FIT		FIT + PRED(25)		FIT + PRED(50)		FIT + PRED(75)		FIT + PRED(90)	
	# of Projects	%	# of Projects	%	# of Projects	%	# of Projects	%	# of Projects	%
COCOMO Performance	28	30%	59	63%	80	86%	91	98%	93	100%
fuzzy-ExCOM Contingency Model Performance	58	62%	78	84%	88	95%	92	99%	93	100%
Change	**30**	**32%**	**18**	**20%**	**8**	**9%**	**0**	**1%**	**0**	**0%**

When using the Fuzzy-ExCOM Model, FIT project for NASA'93 data set improved by factor 32% in comparison to the value of COCOMO. This improvement was also found in projects with categories FIT+PRED(25), FIT+PRED(50), and FIT+PRED(50) by 20%, 9%, and 1% respectively.

From an examination of Table 4.11, we can conclude that the Fuzzy-ExCOM Effort Contingency Allowance Model is capable of providing better estimation than the COCOMO-II Model for the NASA'93 data set.

The result of Contingency Allowance Model applied to TURKISH and INDUSTRY data set are shown in Figure 4.15 and Table 4.14

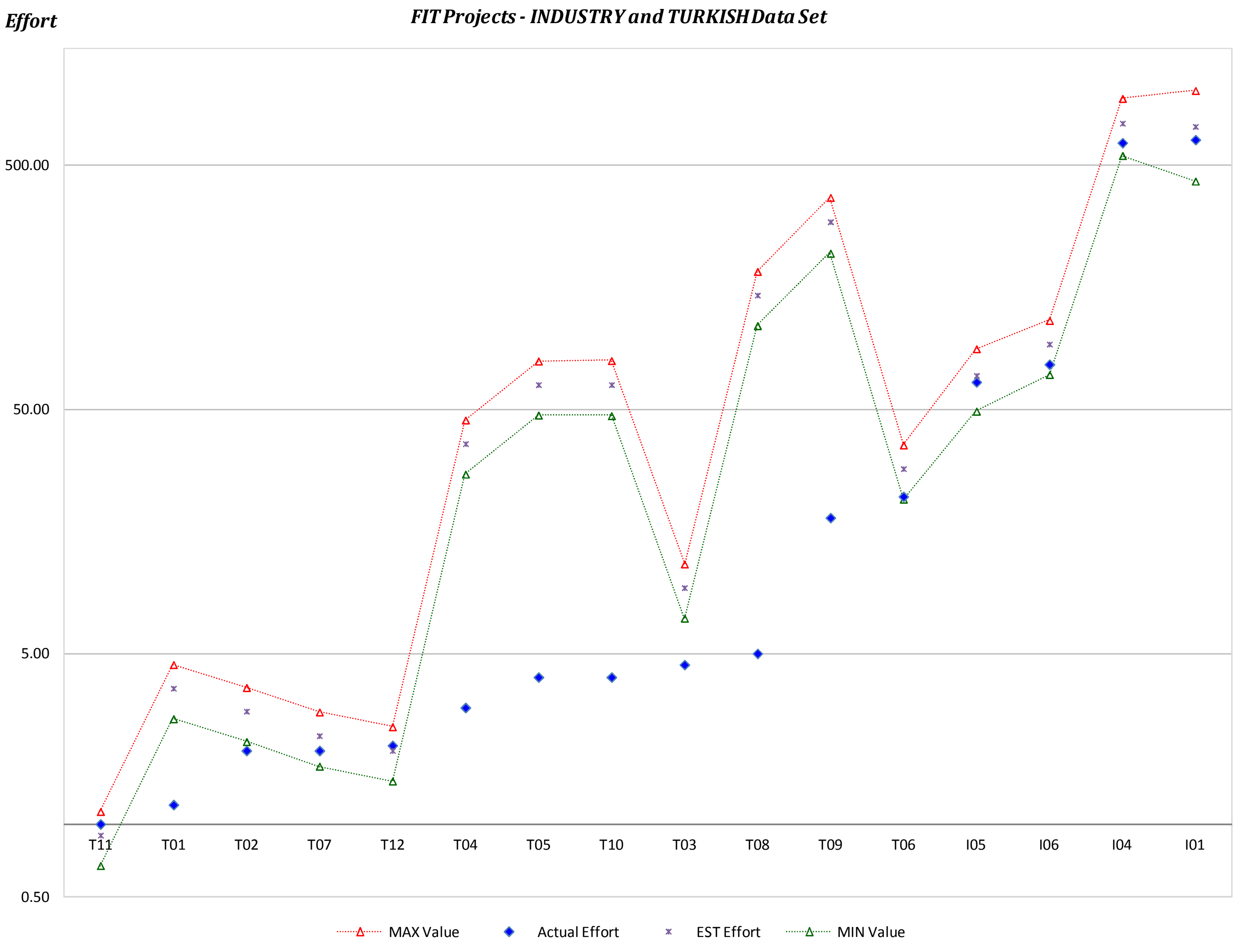


Figure 4.15 FIT Projects (INDUSTRY and TURKISH Data Set)

Table 4.14 FIT Project List (INDUSTRY and TURKISH Data Set)

Proj. ID.	Size (KSLOC)	Size Category	Project Risk	Risk Category	COCOMO II	Contingency Allowance	Contingency Category	MIN Effort Est Value	MAX Effort Est Value	Actual Effort (staff-mo)
I01	196.60	Large	4.49	Low	722.70	40.3%	Medium	431.45	1013.95	638.00
I04	131.00	Large	5.24	Moderate	745.20	26.3%	Medium	549.21	941.19	619.90
I05	13.30	Small	6.32	Moderate	68.90	28.6%	Medium	49.19	88.61	64.80
I06	19.90	Small	4.97	Low	92.70	25.0%	Low	69.53	115.88	76.60
T01	3.00	Small	3.68	Low	3.60	25.0%	Low	2.70	4.50	1.20
T02	2.00	Small	4.01	Low	2.90	25.0%	Low	2.18	3.63	2.00
T03	4.25	Small	4.16	Low	9.30	25.0%	Low	6.98	11.63	4.50
T04	10.00	Small	3.94	Low	36.20	25.0%	Low	27.15	45.25	3.00
T05	15.00	Small	4.58	Low	63.20	25.0%	Low	47.40	79.00	4.00
T06	40.53	Small	4.70	Low	28.60	25.0%	Low	21.45	35.75	22.00
T07	40.50	Small	4.94	Low	2.30	25.0%	Low	1.73	2.88	2.00
T08	31.85	Small	4.79	Low	147.10	25.0%	Low	110.33	183.88	5.00
T09	114.28	Medium	5.18	Moderate	294.00	25.7%	Medium	218.44	369.56	18.00
T10	23.11	Small	5.10	Moderate	63.20	25.4%	Medium	47.15	79.25	4.00
T11	1.37	Small	3.38	Low	0.90	25.0%	Low	0.68	1.13	1.00
T12	1.61	Small	3.95	Low	2.00	25.0%	Low	1.50	2.50	2.10

The performance of Contingency Allowance Model with 5 estimation parameters compared to the performance of COCOMO Model for the industrial project data is described in Table 4.15.

Table 4.15 Contingency Allowance Model and COCOMO Model Performance Comparison (INDUSTRY and TURKISH Data Set)

Total Project = 18	FIT		FIT + PRED(25)		FIT + PRED(50)		FIT + PRED(75)		FIT + PRED(90)	
	# of Projects	%	# of Projects	%	# of Projects	%	# of Projects	%	# of Projects	%
COCOMO Performance	14	78%	18	100%	18	100%	18	100%	18	100%
fuzzy-ExCOM Contingency Model Performance	16	89%	18	100%	18	100%	18	100%	18	100%
Change	2	11%	18	0%	8	0%	0	0%	0	0%

Based on Table 4.15, Fuzzy-ExCOM Model shows the consistent performance in the improvement of the effort estimation activity. FIT projects for INDUSTRY and TURKISH data set (18 project data points) improved by factor 11% in comparison to the value of COCOMO.

4.6 Threats to Validity

As the new approach in the contingency allowance calculation for effort software project estimation based on identified project risks, the development of Fuzzy-ExCOM contains some limitations that can become threat to the model validation such as:

- The soft-computing technique implemented in this model is only the fuzzy technique. Additional soft-computing technique can be investigated to improve the model performance.
- The model is validated using COCOMO public data source and limited industry data. The limited number of data set, especially for industry data set creates the anomaly on correlation calculation results between project risks with software size and actual effort.
- The Effort Contingency Allowance calculation is based only on project risks and software size. Since the software project consider as the most uncertain project compare to other type of project, additional project parameters, such as: project type, development mode, and other possible parameters; can be added to the model in order to improve the accuracy of contingency value.

4.7 Chapter Summary

This chapter describes the Fuzzy-ExCOM Model validation process using three project data sets. The validation results show that Fuzzy-ExCOM Model improves the sensitivity of project risk assessment using Expert COCOMO Methodology and also provides a higher level of effort prediction performance compared to the existing COCOMO effort estimation approach.

The model validation results describe in this chapter provides the answer to RQ-1 (Research Question 1) and RQ-3 (Research Question 3).

Chapter 5
Conclusions and Future Work

The aim of research carried out in this thesis is enhancing the software project planning process by considering the software project risks in effort estimation process. A Fuzzy-ExCOM Model for Software Project Risks assessment and Effort Contingency Allowance is proposed and validated with empirical projects data. This chapter summarizes the conclusions drawn from this research and suggests directions for future work.

5.1 Conclusions

As the most uncertain project compared to the other project, software project development require the sophisticated methods in helping project manager to reduce and manage uncertainty in project execution. Some research describes that the most critical phase in software development project is project planning phase, because the activities that responsible in software project failures are highly related to the project planning phase, such as goal setting, scheduling, staffing, and risk management.

Two main activities in software project planning are effort estimation and risk assessment. Unlike the effort estimation which has almost become the key requirement in project planning, software risk assessment is rarely found and often difficult to implement because of the scarcity of experts, the unique project characteristics, the lack of sufficient time to do a thorough analysis, and being perceived as effort intensive and costly activities.

Software project effort estimation and project risk assessment are integral activities in software project planning phase because the effort estimation accuracy is greatly influenced by the project risks that are inherent in software project. However, most of the software effort estimation methodologies rarely consider project risks in their calculation, even effort estimation and risk management activities in software project planning are often disconnected with each other.

In response to the above problems, the research describes in this thesis introduces a new model called by Fuzzy-ExCOM Model which has the following characteristics and capabilities:

- Improved the risk assessment results using Expert-COCOMO by utilizing the fuzzy techniques to overcome inputs in the form of linguistic terms.

- Improved project planning process by integrating the Effort Estimation activity and Risk Assessment activity in software development project.

- Improved effort estimation results by providing Effort Contingency Allowance that is based on software project risks and software size.

Validation with three project data sets in this research found that the Fuzzy-ExCOM model improves the COCOMO Effort Estimation results and provided answers to those three research questions as follow:

- **RQ-1:** How fuzzy techniques improve the sensitivity of software project risk assessment using Expert-COCOMO methodology?

  Answer: The model validation on Fuzzy-ExCOM Risk Model in Chapter 4 shows that fuzzy technique implemented on Expert-COCOMO methods provides better risk assessment results with a higher level of sensitivity with respect to risk identification compared to the original approach.

- **RQ-2:** How the identified project risks will affect the COCOMO effort estimation approach?

  Answer: As describes in Chapter 2, the ideal estimation value provide base-value with the contingency value to cover the risks and assumption in effort estimation calculation. The identified risk from Fuzzy-ExCOM Risk Model can be used to calculate the contingency value for COCOMO effort estimation using Fuzzy-ExCOM Contingency Model as describes in Chapter 3.

- **RQ-3:** How the identified software project risks will improves the COCOMO effort estimation approach?

  Answer: The model validation in Chapter 4 shows that the effort contingency allowance value calculated using Fuzzy-ExCOM Contingency Model provides a higher level of effort prediction performance compared to the existing COCOMO effort estimation approach.

Overall conclusion from the research stated that Fuzzy-ExCOM Model can be used to complement COCOMO effort estimation by providing the preliminary software project risk assessment based on cost factors, and calculate the contingency allowance to compensate the identified risks.

5.2 Directions for future work

Some limitation in the development of Fuzzy-ExCOM Model as describe in Chapter 4 can be used as the basis for next exciting research in this topic area of software estimation and risk management that may includes:

- The introduction of additional soft computing technique such as ANN (artificial neural-network) to develop the learning ability of the model.

- The Fuzzy-ExCOM Model calibration and improvement can be done by validate the model using additional industrial and public data sets.

- The introduction of additional inputs other than project risks and software size, such as: project type, development mode, and other possible parameters, to improve the Contingency Allowance model performance.

- The implementation of Fuzzy-ExCOM model with other software effort estimation methodology such as SEER-SEM, SLIM, or FP estimation model.

REFERENCES

[1]. The Standish Group International, Inc., CHAOS Summary 2009 - The 10 Laws of Chaos, 2009.

[2]. Marchewka J. T., *Information Technology Project Management*, 3rd Edition, John Wiley & Sons, Inc., 2009, pp.7.

[3]. *Five reason why software projects fail*, Computerworld, May 20, 2002, http://www.computerworld.com/s/article/71209/Why_Projects_Fail, access on Jan 20, 2012.

[4]. "Why Software Fails", *IEEE Spectrum*, September 2005, http://spectrum.ieee.org/computing/software/why-software-fails/3, access on Jan 20, 2012.

[5]. PMI (Project Management Institute), *A Guide to the Project Management Body of Knowledge (PMBOOK Guide)*, 4th Ed, 2008.

[6]. Heldman K. and Heldman W., *CompTIA Project+: Study Guide*, Wiley Publishing Inc., 2010, pp.34.

[7]. Pressman R. S., *Software Engineering – Practitioner's Approach*, 6th Ed, McGraw Hill, NY, 2005, pp.644.

[8]. Huang X., Ho D., Ren J., Capretz L. F., "A Neuro-Fuzzy Tool for Software Estimation", 20th *IEEE International Conference on Software Maintenance*, Chicago, 2004.

[9]. Hall E. M., *Managing Risk: Methods for Software Systems Development*, Addison Wesley Longman, Inc., 1998, pp.39-40.

[10]. Boehm B. W., *Software Engineering Economics*, Prentice Hall, Englewood Cliffs, NJ, 1981.

[11]. Galorath D. D. and Evans M. W., *Software Sizing, Estimation, and Risk Management*, Auerbach Publication, 2006.

[12]. Putnam L. H. and Myers W., *Measures for Excellence*, Prentice Hall, Englewood Cliffs, NJ, 1992.

[13]. Boehm B. W., "Software Risk Management: Principles and Practices", *IEEE Software*, Vol.8, Issue.1, Jan 1991, pp.32-41.

[14]. Jones C., *"Assessment and Control of Software Risks"*, Yourdon Press Prentice Hall, 1994.

[15]. Karolak D. W., *Software Engineering Risk Management*, IEEE Computer Society Press, 1996.

[16]. Charette R. N., *"Software Engineering Risk Analysis and Management"*, McGraw Hill, 1989.

[17]. Hall E. M., *"Managing Risk: Methods for Software Systems Development"*, Addison Wesley, 1998.

[18]. Carr M., et al, *Taxonomy–based Risk Identification*, SEI Technical Report SEI-93-TR-006, Pittsburgh PA, 1993.

[19]. Madachy R., "Heuristic Risk Assessment Using Cost Factors", *IEEE Software*, May/June, Vol.14 No.3, 1997, pp.51-59.

[20]. Odzaly E. E., Greer D., Sage P., "Software Risk Management Barriers: en Empirical Study", *Proceeding 3rd International Symposium on Empirical Software Engineering and Measurement*, 2009, pp.418-421.

[21]. Xia W., Ho D., Capretz L. F., "A neuro-fuzzy model for function point calibration", *WSEAS Transactions on Information Science and Applications 5 (1)*, 2008, pp.22-30.

[22]. Kansala K., "Integrating Risk Assessment with Cost Estimation", *IEEE Software*, Vol14, Issue.3, 1997, pp.61-67.

[23]. Gupta D. and Sadiq M., "Software Risk Assessment and Estimation Model", *Proceeding International Conference on Computer Science and Information Technology*, 2008, pp.963-967.

[24]. Huang S. J., Lin C. Y., Chiu N. H., "Fuzzy Decision Tree Approach for Embedding Risk Assessment Information into Software Estimation Model", *Journal of Information Science and Engineering Vol.22*, 2006, pp.297-313.

[25]. Project Management Institute. http://www.pmi.org/About-Us.aspx, access on Apr 10, 2012.

[26]. Larson E. W. and Gray C. F., *Project Management – The Managerial Process*, McGraw Hill Inc., 2011, pp.3-15.

[27]. Marchewka J. T., *Information Technology Project Management*, 3rd Edition, John Wiley & Sons, Inc., 2009, pp.25.

[28]. IEEE, *Guide to the Software Engineering Body of Knowledge – SWEBOK*, Version 2004.

[29]. Laplante P., *What Every Engineer Should Know about Software Engineering*, CRC Press, Taylor & Francis Group, 2007, pp.1.

[30]. Stellman A. and Greene J., *Applied Software Project Management*, O'Reilly Media, Inc., 2006.

[31]. Marchewka J. T., *Information Technology Project Management*, 3rd Edition, John Wiley & Sons, Inc., 2009, pp.160.

[32]. Fairley R.E., *Managing and Leading Software Projects*, John Wiley & Sons, Inc., 2009, pp. 222.

[33]. Nassif A. B., Ho D., Capretz L. F., "Towards an Early Software Estimation Using Log-Linear Regression and a Multilayer Perceptron Model," *Journal of Systems and Software*, 2012. DOI: 10.1016/j.jss.2012.07.050…

[34]. Mendes E., Mosley N., Watson I., "A comparison of case-based reasoning approaches", *Proceedings of the 11th International Conference on World Wide Web. Honolulu*, Hawaii, USA, 2002, pp. 272–280.

[35]. Du W. L., Ho D., Capretz L. F., "Improving Software Effort Estimation Using Neuro-Fuzzy Model with SEER-SEM", *Global Journal of Computer Science and Technology*, Vol. 10, Issue 12, Oct 2010, pp.52-64.

[36]. Huang X., Ho D., Capretz L. F., "Improving COCOMO model using a neuro-fuzzy approach", *Applied Soft Computing* ,Vol.7, 2007, pp.29-40.

[37]. Attarzadeh I. and Ow S. H., "Proposing a New Software Cost Estimation Model based on Artificial Neural Networks", *2nd International Conference on Computer Engineering and Technology (ICCET)*, 2010.

[38]. Pressman R. S., *Software Engineering – A Practitioner's Approach*, 6th Ed, McGraw Hill, 2005, pp.660-661.

[39]. Oxford Online Dictionary, www.oxforddictionaries.com/definition/estimation?q=estimation, access on Apr 25 2012.

[40]. Coombs P., *IT Project Estimation – A Practical Guide to the Costing of Software,* Cambridge University Press, 2003, pp.33.

[41]. Hart D. H., *Managing the Contingency Allowance*, AIA Best Practices, September 2007.

[42]. Boehm B. W., et al, *Software Cost Estimation with COCOMO II*, Prentice Hall PTR, 2000.

[43]. Iranmanesh S.H., et al, "Risk Assessment of Software Project Using Fuzzy Inference System", *Proceeding International Conference on Computers & Industrial Engineering*, 2009, pp.1149-1154.

[44]. AIRMIC, Alarm, IRM, *A Risk Management Standard*, 2002.

[45]. AIRMIC, Alarm, IRM, *A Structured Approach to Enterprise Risk Management (ERM) and the requirements of ISO 31000*, 2010.

[46]. International Standard, *ISO31000 – Risk Management – Principles and Guidelines*, 1st Ed, 2009.

[47]. PMI (Project Management Institute), *A Guide to the Project Management Body of Knowledge (PMBOOK Guide)*, 4th Ed, 2008, pp.275.

[48]. IEEE Std 1540-2001, *IEEE Standard for Software Life Cycle Processes – Risk Management*, IEEE, 2001.

[49]. Higuera R. P. and Haimes Y. Y., *Software Risk Management – Technical Report CMU/SEI-96-TR-012*, Software Engineering Institute, June 1996.

[50]. Mulcahy R., *"Risk Management – Tricks of the Trade for Project Managers"*, 2nd Ed., RMC Publication Inc., 2010, pp.164.

[51]. Larson E. W. and Gray C. F., *Project Management – The Managerial Process*, McGraw Hill Inc., 2011, pp.584.

[52]. Stepanek G., *"Software Project Secret: Why Software Project Fail"*, Apress Publishing, Sep 2005, pp.192.

[53]. Boehm B. W., *Software Engineering Economics*, Prentice-Hall, Inc., 1981, pp. 278.

[54]. Boehm B. W., "Verifying and Validating Software Requirements and Design Specifications", *IEEE Software*, Jan 1984, pp.75-88.

[55]. Pressman R. S., *Software Engineering – A Practitioner's Approach*, 6^{th} Ed, McGraw Hill, 2005, pp.56.

[56]. Boehm B. W., "Software Risk Management: Principles and Practices", *IEEE Software*, Jan 1991, pp.32-41.

[57]. Boehm B. W., *Tutorial: Software Risk Management*, IEEE Computer Society Press, 1989, pp.115-132.

[58]. Jones C., "*Assessment and Control of Software Risks"*, Yourdon Press Prentice Hall, 1994, pp.47-60.

[59]. Jones C., *"Early Sizing and Early Risk Analysis of Software Project"*, Capers Jones & Associates LLC, January 6, 2012.

[60]. Karolak D. W., *"Software Engineering Risk Management"*, IEEE Computer Society Press, 1996, pp.42.

[61]. Karolak D. W., *"Software Engineering Risk Management"*, IEEE Computer Society Press, 1996, pp.124.

[62]. Charette R. N., *"Managing Risk in Software Maintenance"*, IEEE Software, 1997, pp.43-50.

[63]. Hall E. M., *"Managing Risk: Methods for Software Systems Development"*, Addison Wesley, 1998, pp.7.

[64]. Hall E. M., *"Managing Risk: Methods for Software Systems Development"*, Addison Wesley, 1998, pp.8.

[65]. Hall E. M., *"Managing Risk: Methods for Software Systems Development"*, Addison Wesley, 1998, pp.31.

[66]. Nogueira J. C., *"A Formal Model for Risk Assessment in Software Project"*, Thesis, Naval Postgraduate School, Monterey, California, 2000, pp.23.

[67]. Software Engineering Institute, http://www.sei.cmu.edu/solutions/solutions.cfm, access on March 2012.

[68]. Hall E. M., *"Managing Risk: Methods for Software Systems Development"*, Addison Wesley, 1998, pp.52.

[69]. Stern R. and Arias J. C., "Review of Risk Management Methods", *Business Intelligence Journal*, Vol4No1, January 2011, pp.59-78.

[70]. http://sunset.usc.edu/research/COCOMOII/expert_cocomo/expert_cocomo2000.html, access on Nov 2011.

[71]. Zadeh L. A., "Fuzzy Logic, Neural Networks, and Soft Computing", *Communication of the ACM*, March 1994/Vol.37.No.3, pp.77-84.

[72]. Zadeh L. A., "Fuzzy Sets", *Information & Control Volume 8*, Elsevier, 1965, pp. 338-353.

[73]. Jang J.-S. R., et al, *"Neuro-Fuzzy and Soft Computing, A Computational Approach to Learning and Machine Intelligence"*, Prentice-Hall, Inc., 1997, pp.72.

[74]. Shin Y. C. and Xu C., *"Intelligent Systems: Modeling, Optimization, and Control"*, CRC Press, Taylor & Francis Group, LLC, 2009, pp.22-26.

[75]. Sumathi S. and Surekha P., *"Computational Intelligence Paradigms Theory & Applications using MATLAB"*, CRC Press, 2010, pp.261.

[76]. *"Fuzzy Logic Toolbox User's Guide R2012a"*, The MathWorks, Inc., 1995-2012.

[77]. Xu J., Ho D., Capretz L. F., "An empirical validation of object-oriented design metrics for fault prediction", *Journal of Computer Science 4 (7)*, 2008, pp.571-577.

[78]. Kreinovich V., Quintana C., Reznik L., "Gaussian membership functions are most adequate in representing uncertainty in measurements", *Proceeding NAFIPS: North Amer. Fuzzy Inform. Processing Soc. Conf.*, Puerto Vallarta, Mexico, Dec. 15–17, 1992, vol. II, pp.618–624.

[79]. Coombs P., *IT Project Estimation – A Practical Guide to the Costing of Software,* Cambridge University Press, 2003, pp.3.

[80]. Boehm B. W., *"Software Engineering Economics"*, Prentice Hall, Englewood Cliffs, NJ, 1981, pp.76.

[81]. Boehm B.W., *"Software Engineering Economics"*, Prentice Hall, Englewood Cliffs, NJ, 1981, pp.32.

[82]. Brase C. H. and Brase P. B., *"Understandable Statistics – Concepts and Methods"*, Houghton Mifflin Company, 2003, pp.583.

[83]. PROMISE Software Engineering Repository-NASA93 COCOMO data set - http://promise.site.uottawa.ca/ SERepository/datasets-page.html. Access on May 30, 2012

[84]. PROMISE Software Engineering Repository-Turkish Software Industry data set - http://promisedata.org/ repository/data/cocomo_sdr/cocomo_sdr.arff-Access on May 30, 2012.

[85]. Du W. L.,*"A Neuro-fuzzy Model with SEER-SEM for software effort estimation"*, MESc Thesis, University of Western Ontario, Canada, 2009.

[86]. Center for Systems and Software Engineering, USC. COCOMO II - Constructive Cost Model http://diana.nps.edu/~madachy/tools/COCOMOII.php. Access on June 2, 2012.

Appendix A: NASA COCOMO Data Points

```
%%-*- text -*-
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This is a PROMISE Software Engineering Repository data set made publicly
% available in order to encourage repeatable, verifiable, refutable, and/or
% improvable predictive models of software engineering.
%
% If you publish material based on PROMISE data sets then, please
% follow the acknowledgment guidelines posted on the PROMISE repository
% web page http://promise.site.uottawa.ca/SERepository .
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% 1. Title/Topic: COCOMO NASA 2 / Software cost estimation

% 2. Sources:
%
%     -- 93 NASA projects from different centers
%        for projects from the following years:
%
%    n year
%  --- ----
%    1 1971
%    1 1974
%    2 1975
%    2 1976
%   10 1977
%    4 1978
%   19 1979
%   11 1980
%   13 1982
%    7 1983
%    7 1984
%    6 1985
%    8 1986
%    2 1987
%
%    Collected by
%        Jairus Hihn, JPL, NASA, Manager SQIP Measurement &
%          Benchmarking Element
%        Phone (818) 354-1248 (Jairus.M.Hihn@jpl.nasa.gov)
%
%     -- Donor: Tim Menzies (tim@menzies.us)
%
%     -- Date: Feb 8 2006
%
% 3. Past Usage
%     None with this specific data set. But for older work on similar data, see:
%
%     1. "Validation Methods for Calibrating Software Effort
%         Models", T. Menzies and D. Port and Z. Chen and
%         J. Hihn and S. Stukes, Proceedings ICSE 2005,
%         http://menzies.us/pdf/04coconut.pdf
%         -- Results
%             -- Given background knowledge on 60 prior projects,
%                a new cost model can be tuned to local data using
%                as little as 20 new projects.
%             -- A very simple calibration method (COCONUT) can
%                achieve PRED(30)=7% or PRED(20)=50% (after 20 projects).
%                These are results seen in 30 repeats of an incremental
%                cross-validation study.
%             -- Two cost models are compared; one based on just
%                lines of code and one using over a dozen "effort
%                multipliers". Just using lines of code loses 10 to 20
%                PRED(N) points.
%
```

```
%  3.1 Additional Usage:
%    2. "Feature Subset Selection Can Improve Software Cost Estimation Accuracy"
%         Zhihao Chen, Tim Menzies, Dan Port and Barry Boehm
%         Proceedings PROMISE Workshop 2005,
%         http://www.etechstyle.com/chen/papers/05fsscocomo.pdf
%         P02, P03, P04 are used in this paper.
%         -- Results
%             -- To the best of our knowledge, this is the first report
%                of applying feature subset selection (FSS)
%                to software effort data.
%
%             -- FSS can dramatically improve cost estimation.
%
%             ---T-tests are applied to the results to demonstrate
%                that always in our data sets, removing
%                attributes improves performance without increasing the
%                variance in model behavior.
%

% 4. Relevant Information
%
%    The COCOMO software cost model measures effort in calendar months
%    of 152 hours (and includes development and management hours).
%    COCOMO assumes that the effort grows more than linearly on
%    software size; i.e. months=a* KSLOC^b*c. Here, "a" and "b" are
%    domain-specific parameters; "KSLOC" is estimated directly or
%    computed from a function point analysis; and "c" is the product
%    of over a dozen "effort multipliers". I.e.
%
%    months=a*(KSLOC^b)*(EM1* EM2 * EM3 * ...)
%
%    The effort multipliers are as follows:
%
%     increase | acap | analysts capability
%     these to | pcap | programmers capability
%    decrease  | aexp | application experience
%      effort  | modp | modern programing practices
%              | tool | use of  software tools
%              | vexp | virtual machine experience
%              | lexp | language experience
%    ----------+------+--------------------------
%              | sced | schedule constraint
%    ----------+------+--------------------------
%     decrease | stor | main memory constraint
%     these to | data | data base size
%     decrease | time | time constraint for cpu
%       effort | turn | turnaround time
%              | virt | machine volatility
%              | cplx | process complexity
%              | rely | required software reliability
%
%    In COCOMO I, the exponent on KSLOC was a single value ranging from
%    1.05 to 1.2.  In COCOMO II, the exponent "b" was divided into a
%    constant, plus the sum of five "scale factors" which modeled
%    issues such as ``have we built this kind of system before?''.  The
%    COCOMO~II effort multipliers are similar but COCOMO~II dropped one
%    of the effort multiplier parameters; renamed some others; and
%    added a few more (for "required level of reuse", "multiple-site
%    development", and "schedule pressure").
%
%    The effort multipliers fall into three groups: those that are
%    positively correlated to more effort; those that are
%    negatively correlated to more effort; and a third group
%    containing just schedule information. In COCOMO~I, "sced" has a
%    U-shaped correlation to effort; i.e. giving programmers either
%    too much or too little time to develop a system can be
%    detrimental.
%
%    The numeric values of the effort multipliers are:
%
%                very                                       very      extra     productivity
```

```
%                low         low       nominal   high      high      high      range
%     ----------------------------------------------------------------------
%     acap 1.46      1.19      1.00      0.86      0.71                2.06
%     pcap 1.42.     1.17      1.00      0.86      0.70                1.67
%     aexp           1.29      1.13      1.00      0.91      0.82                1.57
%     modp           1.24.     1.10      1.00      0.91      0.82                1.34
%     tool           1.24      1.10      1.00      0.91      0.83                1.49
%     vexp           1.21      1.10      1.00      0.90                          1.34
%     lexp           1.14      1.07      1.00      0.95                          1.20
%     sced           1.23      1.08      1.00      1.04      1.10                e
%     stor                               1.00      1.06      1.21      1.56      -1.21
%     data                      0.94     1.00      1.08      1.16                -1.23
%     time                               1.00      1.11      1.30      1.66      -1.30
%     turn                     0.87      1.00      1.07      1.15                -1.32
%     virt                     0.87      1.00      1.15      1.30                -1.49
%     rely           0.75       0.88      1.00      1.15      1.40               -1.87
%     cplx           0.70      0.85      1.00      1.15      1.30      1.65      -2.36
%
%     These were learnt by Barry Boehm after a regression analysis of the
%     projects in the COCOMO I data set.
%       @Book{boehm81,
%         Author    =          "B. Boehm",
%         Title     =          "Software Engineering Economics",
%         Publisher =          "Prentice Hall",
%         Year      =          1981}
%
%      The last column of the above table shows max(E)/min(EM) and shows
%      the overall effect of a single effort multiplier. For example,
%      increasing "acap" (analyst experience) from very low to very
%      high will most decrease effort while increasing "rely"
%      (required reliability) from very low to very high will most
%      increase effort.
%
%      There is much more to COCOMO that the above description. The
%      COCOMO~II text is over 500 pages long and offers
%      all the details needed to implement data capture and analysis of
%      COCOMO in an industrial context.
%        @Book{boehm00b,
%          Author = "Barry Boehm and Ellis Horowitz and Ray Madachy and
%                    Donald Reifer and Bradford K. Clark and Bert Steece
%                    and A. Winsor Brown and Sunita Chulani and Chris Abts",
%           Title = "Software Cost Estimation with Cocomo II",
%       Publisher = "Prentice Hall",
%            Year = 2000,
%            ibsn = "0130266922"}
%
%      Included in that book is not just an effort model but other
%      models for schedule, risk, use of COTS, etc.  However, most
%      (?all) of the validation work on COCOMO has focused on the effort
%      model.
%        @article{chulani99,
%          author =  "S. Chulani and B. Boehm and B. Steece",
%          title =   "Bayesian Analysis of Empirical Software Engineering
%                           Cost Models",
%          journal = "IEEE Transaction on Software Engineering",
%          volume =  25,
%          number =  4,
%          month =   "July/August",
%          year =    "1999"}
%
%      The value of an effort predictor can be reported many ways
%      including MMRE and PRED(N).MMRE and PRED are computed from the
%      relative error, or RE, which is the relative size of the
%      difference between the actual and estimated value:
%
%      RE.i = (estimate.i - actual.i) / (actual.i)
%
%      Given a data set of of size "D", a "Train"ing set of size
%      "(X=|Train|) <= D", and a "test" set of size "T=D-|Train|", then
%      the mean magnitude of the relative error, or MMRE, is the
%      percentage of the absolute values of the relative errors,
```

```
%     averaged over the "T" items in the "Test" set; i.e.
%
%     MRE.i  = abs(RE.i)
%     MMRE.i = 100/T*( MRE.1 + MRE.2 + ... + MRE.T)
%
%     PRED(N) reports the average percentage of estimates that were
%     within N% of the actual values:
%
%     count=0
%     for(i=1;i<=T;i++) do if (MRE.i <= N/100) then count++ fi done
%     PRED(N) = 100/T * sum
%
%     For example, e.g. PRED(30)=50% means that half the estimates are
%     within 30% of the actual.  Shepperd and Schofield comment that
%     "MMRE is fairly conservative with a bias against overestimates
%     while Pred(25) will identify those prediction systems that are
%     generally accurate but occasionally wildly inaccurate".
%       @article{shepperd97,
%         author="M. Shepperd and C. Schofield",
%         title="Estimating Software Project Effort Using Analogies",
%         journal="IEEE Transactions on Software Engineering",
%         volume=23,
%         number=12,
%         month="November",
%         year=1997,
%         note="Available from
%                \url{http://www.utdallas.edu/~rbanker/SE_XII.pdf}"}
%

% 5. Number of instances: 93

% 6. Number of attributes: 24
%       - 15 standard COCOMO-I discrete  attributes in the range Very_Low to
%         Extra_High
%       - 7 others describing the project;
%       - one lines of code measure,
%       - one goal field being the actual effort in person months.

% 7. Attribute information:

@relation cocomonasa_2

% Unique id
@attribute recordnumber real % unique id. note that these numbers are
                                  % NOT continguous since the records shown
                                  % below are a subset of another NASA
                                  % database.

%project name
@attribute projectname {de,erb,gal,X,hst,slp,spl,Y}

%cagetory of application
@attribute cat2 {Avionics, application_ground, avionicsmonitoring, batchdataprocessing,
communications, datacapture, launchprocessing, missionplanning, monitor_control,
operatingsystem, realdataprocessing, science, simulation, utility}

% flight or ground system?
@attribute forg {f,g}

%which nasa center?
@attribute center {1,2,3,4,5,6}

%year of development
@attribute year real

%development mode
@attribute mode {embedded,organic,semidetached}

%cocomo attributes: described above in section 4
@attribute rely {vl,l,n,h,vh,xh}
@attribute data {vl,l,n,h,vh,xh}
```

```
@attribute cplx {vl,l,n,h,vh,xh}
@attribute time {vl,l,n,h,vh,xh}
@attribute stor {vl,l,n,h,vh,xh}
@attribute virt {vl,l,n,h,vh,xh}
@attribute turn {vl,l,n,h,vh,xh}
@attribute acap {vl,l,n,h,vh,xh}
@attribute aexp {vl,l,n,h,vh,xh}
@attribute pcap {vl,l,n,h,vh,xh}
@attribute vexp {vl,l,n,h,vh,xh}
@attribute lexp {vl,l,n,h,vh,xh}
@attribute modp {vl,l,n,h,vh,xh}
@attribute tool {vl,l,n,h,vh,xh}
@attribute sced {vl,l,n,h,vh,xh}

%equivalent physical 1000 lines of source code
@attribute equivphyskloc real

%development effort in months (one month =152 hours and includes development and management
hours)
@attribute act_effort real

% Section 8. Missing attributes: none

% Section 9: Distribution of class values
%
%  #  development months
%  == ==================
%  46    0 - 499
%  28  500 - 999
%   7 1000 - 1499
%   3 1500 - 1999
%   3 2000 - 2499
%   3 2500 - 2999
%   0 3000 - 3999
%   1 4000 - 4499
%   1 4500 - 4999
%   0 5000 - 7999
%   1 8000

@data

1,de,avionicsmonitoring,g,2,1979,semidetached,h,l,h,n,n,l,l,n,n,n,n,h,h,n,l,25.9,117.6
2,de,avionicsmonitoring,g,2,1979,semidetached,h,l,h,n,n,l,l,n,n,n,n,h,h,n,l,24.6,117.6
3,de,avionicsmonitoring,g,2,1979,semidetached,h,l,h,n,n,l,l,n,n,n,n,h,h,n,l,7.7,31.2
4,de,avionicsmonitoring,g,2,1979,semidetached,h,l,h,n,n,l,l,n,n,n,n,h,h,n,l,8.2,36
5,de,avionicsmonitoring,g,2,1979,semidetached,h,l,h,n,n,l,l,n,n,n,n,h,h,n,l,9.7,25.2
6,de,avionicsmonitoring,g,2,1979,semidetached,h,l,h,n,n,l,l,n,n,n,n,h,h,n,l,2.2,8.4
7,de,avionicsmonitoring,g,2,1979,semidetached,h,l,h,n,n,l,l,n,n,n,n,h,h,n,l,3.5,10.8
8,erb,avionicsmonitoring,g,2,1982,semidetached,h,l,h,n,n,l,l,n,n,n,n,h,h,n,l,66.6,352.8
9,gal,missionplanning,g,1,1980,semidetached,h,l,h,xh,xh,l,h,h,h,h,n,h,h,h,n,7.5,72
10,gal,missionplanning,g,1,1980,semidetached,n,l,h,n,n,l,l,h,vh,vh,n,h,n,n,n,20,72
11,gal,missionplanning,g,1,1984,semidetached,n,l,h,n,n,l,l,h,vh,h,n,h,n,n,n,6,24
12,gal,missionplanning,g,1,1980,semidetached,n,l,h,n,n,l,l,h,vh,vh,n,h,n,n,n,100,360
13,gal,missionplanning,g,1,1985,semidetached,n,l,h,n,n,l,l,h,vh,n,n,l,n,n,n,11.3,36
14,gal,missionplanning,g,1,1980,semidetached,n,l,h,n,n,h,l,h,h,h,l,vl,n,n,n,100,215
15,gal,missionplanning,g,1,1983,semidetached,n,l,h,n,n,l,l,h,vh,h,n,h,n,n,n,20,48
16,gal,missionplanning,g,1,1982,semidetached,n,l,h,n,n,l,l,h,n,n,n,vl,n,n,n,100,360
17,gal,missionplanning,g,1,1980,semidetached,n,l,h,n,xh,l,l,h,vh,vh,n,h,n,n,n,150,324
18,gal,missionplanning,g,1,1984,semidetached,n,l,h,n,n,l,l,h,h,h,n,h,n,n,n,31.5,60
19,gal,missionplanning,g,1,1983,semidetached,n,l,h,n,n,l,l,h,vh,h,n,h,n,n,n,15,48
20,gal,missionplanning,g,1,1984,semidetached,n,l,h,n,xh,l,l,h,h,n,n,h,n,n,n,32.5,60
21,X,avionicsmonitoring,g,2,1985,semidetached,h,l,h,n,n,l,l,n,n,n,n,h,h,n,l,19.7,60
22,X,avionicsmonitoring,g,2,1985,semidetached,h,l,h,n,n,l,l,n,n,n,n,h,h,n,l,66.6,300
23,X,simulation,g,2,1985,semidetached,h,l,h,n,n,l,l,n,n,n,n,h,h,n,l,29.5,120
24,X,monitor_control,g,2,1986,semidetached,h,n,n,h,n,n,n,n,h,h,n,n,n,n,n,15,90
25,X,monitor_control,g,2,1986,semidetached,h,n,h,n,n,n,n,n,h,h,n,n,n,n,n,38,210
26,X,monitor_control,g,2,1986,semidetached,n,n,n,n,n,n,n,n,h,h,n,n,n,n,n,10,48
27,X,realdataprocessing,g,2,1982,semidetached,n,vh,h,vh,vh,l,h,vh,h,n,l,h,vh,vh,l,15.4,70
28,X,realdataprocessing,g,2,1982,semidetached,n,vh,h,vh,vh,l,h,vh,h,n,l,h,vh,vh,l,48.5,239
29,X,realdataprocessing,g,2,1982,semidetached,n,vh,h,vh,vh,l,h,vh,h,n,l,h,vh,vh,l,16.3,82
30,X,communications,g,2,1982,semidetached,n,vh,h,vh,vh,l,h,vh,h,n,l,h,vh,vh,l,12.8,62
```

31,X,batchdataprocessing,g,2,1982,semidetached,n,vh,h,vh,vh,l,h,vh,h,n,l,h,vh,vh,l,32.6,170
32,X,datacapture,g,2,1982,semidetached,n,vh,h,vh,vh,l,h,vh,h,n,l,h,vh,vh,l,35.5,192
33,X,missionplanning,g,2,1985,semidetached,h,l,h,n,n,l,l,n,n,n,n,h,h,n,l,5.5,18
34,X,avionicsmonitoring,g,2,1987,semidetached,h,l,h,n,n,l,l,n,n,n,n,h,h,n,l,10.4,50
35,X,avionicsmonitoring,g,2,1987,semidetached,h,l,h,n,n,l,l,n,n,n,n,h,h,n,l,14,60
36,X,monitor_control,g,2,1986,semidetached,h,n,h,n,n,n,n,n,n,n,n,n,n,n,n,6.5,42
37,X,monitor_control,g,2,1986,semidetached,n,n,h,n,n,n,n,n,n,n,n,n,n,n,n,13,60
38,X,monitor_control,g,2,1986,semidetached,n,n,h,n,n,n,n,n,n,h,n,h,h,h,n,90,444
39,X,monitor_control,g,2,1986,semidetached,n,n,h,n,n,n,n,n,n,n,n,n,n,n,n,8,42
40,X,monitor_control,g,2,1986,semidetached,n,n,h,h,n,n,n,n,n,n,n,n,n,n,n,16,114
41,hst,datacapture,g,2,1980,semidetached,n,h,h,vh,h,l,h,h,n,h,l,h,h,n,l,177.9,1248
42,slp,launchprocessing,g,6,1975,semidetached,h,l,h,n,n,l,l,n,n,h,n,n,h,vl,n,302,2400
43,Y,application_ground,g,5,1982,semidetached,n,h,l,n,n,h,n,h,h,n,n,n,h,h,n,282.1,1368
44,Y,application_ground,g,5,1982,semidetached,h,h,l,n,n,n,h,h,h,n,n,n,h,n,n,284.7,973
45,Y,avionicsmonitoring,g,5,1982,semidetached,h,h,n,n,n,l,l,n,h,h,n,h,n,n,n,79,400
46,Y,avionicsmonitoring,g,5,1977,semidetached,l,n,n,n,n,l,l,h,h,vh,n,h,l,l,h,423,2400
47,Y,missionplanning,g,5,1977,semidetached,n,n,n,n,n,l,n,h,vh,vh,l,h,h,n,n,190,420
48,Y,missionplanning,g,5,1984,semidetached,n,n,h,n,h,n,n,h,h,n,n,h,h,n,h,47.5,252
49,Y,missionplanning,g,5,1980,semidetached,vh,n,xh,h,h,l,l,n,h,n,n,n,l,h,n,21,107
50,Y,simulation,g,5,1983,semidetached,n,h,h,vh,n,n,h,h,h,h,n,h,l,l,h,78,571.4
51,Y,simulation,g,5,1984,semidetached,n,h,h,vh,n,n,h,h,h,h,n,h,l,l,h,11.4,98.8
52,Y,simulation,g,5,1985,semidetached,n,h,h,vh,n,n,h,h,h,h,n,h,l,l,h,19.3,155
53,Y,missionplanning,g,5,1979,semidetached,h,n,vh,h,h,l,h,h,n,n,h,h,l,vh,h,101,750
54,Y,missionplanning,g,5,1979,semidetached,h,n,h,h,h,l,h,n,h,n,n,n,l,vh,n,219,2120
55,Y,utility,g,5,1979,semidetached,h,n,h,h,h,l,h,n,h,n,n,n,l,vh,n,50,370
56,spl,datacapture,g,2,1979,semidetached,vh,h,h,vh,vh,n,n,vh,vh,vh,n,h,h,h,l,227,1181
57,spl,batchdataprocessing,g,2,1977,semidetached,n,h,vh,n,n,l,n,h,n,vh,l,n,h,n,l,70,278
58,de,avionicsmonitoring,g,2,1979,semidetached,h,l,h,n,n,l,l,n,n,n,n,h,h,n,l,0.9,8.4
59,slp,operatingsystem,g,6,1974,semidetached,vh,l,xh,xh,vh,l,l,h,vh,h,vl,h,vl,vl,h,980,4560
60,slp,operatingsystem,g,6,1975,embedded,n,l,h,n,n,l,l,vh,n,vh,h,h,n,l,n,350,720
61,Y,operatingsystem,g,5,1976,embedded,h,n,xh,h,h,l,l,h,n,n,h,h,h,h,n,70,458
62,Y,utility,g,5,1979,embedded,h,n,xh,h,h,l,l,h,n,n,h,h,h,h,n,271,2460
63,Y,avionicsmonitoring,g,5,1971,organic,n,n,n,n,n,l,l,h,h,h,n,h,n,l,n,90,162
64,Y,avionicsmonitoring,g,5,1980,organic,n,n,n,n,n,l,l,h,h,h,n,h,n,l,n,40,150
65,Y,avionicsmonitoring,g,5,1979,embedded,h,n,h,h,n,l,l,h,h,h,n,h,n,n,n,137,636
66,Y,avionicsmonitoring,g,5,1977,embedded,h,n,h,h,n,h,l,h,h,h,n,h,n,vl,n,150,882
67,Y,avionicsmonitoring,g,5,1976,embedded,vh,n,h,h,n,l,l,h,h,h,n,h,n,n,n,339,444
68,Y,avionicsmonitoring,g,5,1983,organic,l,h,l,n,n,h,l,h,h,h,n,h,n,l,n,240,192
69,Y,avionicsmonitoring,g,5,1978,semidetached,h,n,h,n,vh,l,n,h,h,h,h,h,l,l,l,144,576
70,Y,avionicsmonitoring,g,5,1979,semidetached,n,l,n,n,vh,l,n,h,h,h,h,h,l,l,l,151,432
71,Y,avionicsmonitoring,g,5,1979,semidetached,n,l,h,n,vh,l,n,h,h,h,h,h,l,l,l,34,72
72,Y,avionicsmonitoring,g,5,1979,semidetached,n,n,h,n,vh,l,n,h,h,h,h,h,l,l,l,98,300
73,Y,avionicsmonitoring,g,5,1979,semidetached,n,n,h,n,vh,l,n,h,h,h,h,h,l,l,l,85,300
74,Y,avionicsmonitoring,g,5,1982,semidetached,n,l,n,n,vh,l,n,h,h,h,h,h,l,l,l,20,240
75,Y,avionicsmonitoring,g,5,1978,semidetached,n,l,n,n,vh,l,n,h,h,h,h,h,l,l,l,111,600
76,Y,avionicsmonitoring,g,5,1978,semidetached,h,vh,h,n,vh,l,n,h,h,h,h,h,l,l,l,162,756
77,Y,avionicsmonitoring,g,5,1978,semidetached,h,h,vh,n,vh,l,n,h,h,h,h,h,l,l,l,352,1200
78,Y,operatingsystem,g,5,1979,semidetached,h,n,vh,n,vh,l,n,h,h,h,h,h,l,l,l,165,97
79,Y,missionplanning,g,5,1984,embedded,h,n,vh,h,h,l,vh,h,n,n,h,h,h,vh,h,60,409
80,Y,missionplanning,g,5,1984,embedded,h,n,vh,h,h,l,vh,h,n,n,h,h,h,vh,h,100,703
81,hst,Avionics,f,2,1980,embedded,h,vh,vh,xh,xh,h,h,n,n,n,l,l,n,n,h,32,1350
82,hst,Avionics,f,2,1980,embedded,h,h,h,vh,xh,h,h,h,h,h,h,h,h,n,n,53,480
84,spl,Avionics,f,3,1977,embedded,h,l,vh,vh,xh,l,n,vh,vh,vh,vl,vl,h,h,n,41,599
89,spl,Avionics,f,3,1977,embedded,h,l,vh,vh,xh,l,n,vh,vh,vh,vl,vl,h,h,n,24,430
91,Y,Avionics,f,5,1977,embedded,vh,h,vh,xh,xh,n,n,h,h,h,h,h,h,n,h,165,4178.2
92,Y,science,f,5,1977,embedded,vh,h,vh,xh,xh,n,n,h,h,h,h,h,h,n,h,65,1772.5
93,Y,Avionics,f,5,1977,embedded,vh,h,vh,xh,xh,n,l,h,h,h,h,h,h,n,h,70,1645.9
94,Y,Avionics,f,5,1977,embedded,vh,h,xh,xh,xh,n,n,h,h,h,h,h,h,n,h,50,1924.5
97,gal,Avionics,f,5,1982,embedded,vh,l,vh,vh,xh,l,l,h,l,n,vl,l,l,h,h,7.25,648
98,Y,Avionics,f,5,1980,embedded,vh,h,vh,xh,xh,n,n,h,h,h,h,h,h,n,h,233,8211
99,X,Avionics,f,2,1983,embedded,h,n,vh,vh,vh,h,h,n,n,n,l,l,n,n,h,16.3,480
100,X,Avionics,f,2,1983,embedded,h,n,vh,vh,vh,h,h,n,n,n,l,l,n,n,h,6.2,12
101,X,science,f,2,1983,embedded,h,n,vh,vh,vh,h,h,n,n,n,l,l,n,n,h,3,38

Appendix B: TURKISH and INDUSTRY Data Set.

Project Id	LOC	Act Eff (man-mo)	Scale Factors					Product Attributes					Platform Attributes			Personnel Attributes						Project Attributes		
			PREC	FLEX	RESL	TEAM	PMAT	RELY	DATA	CPLX	RUSE	DOCU	TIME	STOR	PVOL	ACAP	PCAP	PCON	AEXP	PEXP	LTEX	TOOL	SITE	SCED
T01	3,000	1.2	V-High	Nominal	Low	High	V-Low	Nominal	Nominal	Low	Low	Low	V-High	High	Nominal	High	V-High	V-High	V-High	High	High	High	Low	Nominal
T02	2,000	2	V-High	Nominal	Low	V-High	V-Low	High	Nominal	Low	Nominal	Low	V-High	High	Nominal	High	High	V-High	V-High	V-High	V-High	High	V Low	High
T03	4,250	4.5	E-High	Nominal	Low	High	V-Low	High	V-High	Low	Nominal	Low	V-High	V-High	Nominal	High	High	V-High	V-High	V-High	V-High	High	V Low	High
T04	10,000	3	E-High	High	High	High	Nominal	High	High	V-High	Low	Nominal	Nominal	Nominal	High	High	High	V-Low	V-High	V-High	High	Nominal	High	High
T05	15,000	4	Nominal	Nominal	Low	High	Nominal	Low	Nominal	V-High	Nominal	High	High	Nominal	Nominal	High	High	High	Nominal	Nominal	Nominal	Nominal	High	Low
T06	40,530	22	V-High	Low	E-High	E-High	High	Low	Low	Nominal	Low	V-Low	Nominal	Nominal	Low	V-High	V-High	High	Nominal	High	High	V-Low	E-High	Low
T07	4,050	2	E-High	V-High	E-High	Nominal	High	Low	Low	Low	Nominal	V-Low	Nominal	Nominal	Low	V-High	V-High	V-High	Nominal	V-High	V-High	V-Low	High	Low
T08	31,845	5	Nominal	V-Low	V-High	Nominal	Low	Low	V-High	V-Low	High	V-High	V-High	V-High	Low	Low	High	V-High	Nominal	Nominal	Nominal	Nominal	V-High	Nominal
T09	114,280	18	V-Low	V-Low	Nominal	V-Low	Low	V-Low	Nominal	Low	V-High	Low	Nominal	Nominal	Low	Low	High	V-High	High	High	High	Nominal	High	Nominal
T10	23,106	4	Nominal	V-Low	V-High	Low	Low	V-Low	High	Nominal	V-High	V-Low	V-High	V-High	Low	Low	V-High	V-High	High	V-High	High	Nominal	Nominal	Low
T11	1,369	1	High	Nominal	V-High	High	Low	Low	Nominal	Nominal	Nominal	Low	Nominal	Nominal	Low	V-High	High	V-High	Nominal	Nominal	High	V-High	V-High	High
T12	1,611	2.1	Low	Nominal	Nominal	High	Nominal	Low	Low	High	High	Nominal	Nominal	Nominal	Low	Low	High	V-High	High	Nominal	High	V-High	V-High	High

Project Id	LOC	Act Eff (man-mo)	Scale Factors					Product Attributes					Platform Attributes			Personnel Attributes						Project Attributes		
			PREC	FLEX	RESL	TEAM	PMAT	RELY	DATA	CPLX	RUSE	DOCU	TIME	STOR	PVOL	ACAP	PCAP	PCON	AEXP	PEXP	LTEX	TOOL	SITE	SCED
I1	196,600	638	High	High	High	V-High	Low	Nominal	Low	High	High	High	High	High	Nominal	High	High	Nominal	Nominal	High	High	High	Nominal	Low
I2	51,800	185	High	High	High	V-High	Low	Nominal	Low	High	High	High	Nominal	High	Nominal	High	High	Nominal	High	High	High	High	Nominal	Low
I3	64,100	332	High	High	High	V-High	Low	Nominal	Low	High	High	High	Nominal	High	Nominal	Nominal	Nominal	Nominal	Nominal	High	Nominal	High	Nominal	Nominal
I4	131,000	619.9	High	High	High	V-High	Low	High	Low	Nominal	High	High	Nominal	High	Nominal	Nominal	Nominal	Nominal	Nominal	Low	Low	Nominal	Nominal	Nominal
I5	13,300	64.8	High	High	High	V-High	V-Low	High	Nominal	High	High	High	Nominal	High	Nominal	Nominal	Nominal	Nominal	Nominal	High	Nominal	Nominal	Nominal	Nominal
I6	19,900	76.6	High	High	High	V-High	Low	Nominal	Low	High	High	High	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Low	Low	Low	High	Nominal	Nominal

Appendix C: Risk Rules.

SCED Risk Rules

SCED

CPLX	Very Low	Low	Nominal	High	Very High
Very Low					
Low	Very Low				
Nominal	Low	Very Low			
High	Moderate	Low	Very Low		
Very High	High	Moderate	Low	Very Low	
Extra High	Very High	High	Moderate	Low	Very Low

SCED

AEXP	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				

SCED

RELY	Very Low	Low	Nominal	High	Very High
Very Low	Very Low				
Low	Low	Very Low			
Nominal	Moderate	Low	Very Low		
High	High	Moderate	Low	Very Low	
Very High	Very High	High	Moderate	Low	Very Low

SCED

PCAP	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				

SCED

TIME	Very Low	Low	Nominal	High	Very High
Nominal	Very High	High	Moderate	Low	Very Low
High	High	Moderate	Low	Very Low	
Very High	Moderate	Low	Very Low		
Extra High	Low	Very Low			

SCED

LTEX	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				

SCED

PVOL	Very Low	Low	Nominal	High	Very High
Low	Low	Very Low			
Nominal	Moderate	Low	Very Low		
High	High	Moderate	Low	Very Low	
Very High	Very High	High	Moderate	Low	Very Low

SCED

PMAT	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				
Extra High					

SCED

TOOL	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				

SCED

TEAM	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				
Extra High					

SCED

ACAP	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				

ACAP Risk Rules

		PMAT					
		Very Low	Low	Nominal	High	Very High	Extra High
ACAP	Very Low	Very High	High	Moderate	Low	Very Low	
	Low	High	Moderate	Low	Very Low		
	Nominal	Moderate	Low	Very Low			
	High	Low	Very Low				
	Very High	Very Low					

		TOOL				
		Very Low	Low	Nominal	High	Very High
ACAP	Very Low	Very High	High	Moderate	Low	Very Low
	Low	High	Moderate	Low	Very Low	
	Nominal	Moderate	Low	Very Low		
	High	Low	Very Low			
	Very High	Very Low				

AEXP Risk Rules

		RUSE				
		Low	Nominal	High	Very High	Extra High
AEXP	Very Low	Very High	High	Moderate	Low	Very Low
	Low	High	Moderate	Low	Very Low	
	Nominal	Moderate	Low	Very Low		
	High	Low	Very Low			
	Very High	Very Low				

		TEAM					
		Very Low	Low	Nominal	High	Very High	Extra High
AEXP	Very Low	Very High	High	Moderate	Low	Very Low	
	Low	High	Moderate	Low	Very Low		
	Nominal	Moderate	Low	Very Low			
	High	Low	Very Low				
	Very High	Very Low					

LTEX Risk Rule

		RUSE				
		Low	Nominal	High	Very High	Extra High
LTEX	Very Low	Very High	High	Moderate	Low	Very Low
	Low	High	Moderate	Low	Very Low	
	Nominal	Moderate	Low	Very Low		
	High	Low	Very Low			
	Very High	Very Low				

PCAP Risk Rules

PCAP \ PMAT	Very Low	Low	Nominal	High	Very High	Extra High
Very Low	Very High	High	Moderate	Low	Very Low	
Low	High	Moderate	Low	Very Low		
Nominal	Moderate	Low	Very Low			
High	Low	Very Low				
Very High	Very Low					

PCAP \ TOOL	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				

TOOL Risk Rule

PMAT \ TOOL	Very Low	Low	Nominal	High	Very High
Very Low	Very High	High	Moderate	Low	Very Low
Low	High	Moderate	Low	Very Low	
Nominal	Moderate	Low	Very Low		
High	Low	Very Low			
Very High	Very Low				
Extra High					

SITE Risk Rule

SITE \ TEAM	Very Low	Low	Nominal	High	Very High	Extra High
Very Low	Very High	High	Moderate	Low	Very Low	
Low	High	Moderate	Low	Very Low		
Nominal	Moderate	Low	Very Low			
High	Low	Very Low				
Very High	Very Low					
Extra High						

RELY Risk Rules

ACAP \ RELY	Very Low	Low	Nominal	High	Very High
Very Low	Very Low	Low	Moderate	High	Very High
Low		Very Low	Low	Moderate	High
Nominal			Very Low	Low	Moderate
High				Very Low	Low
Very High					Very Low

PCAP \ RELY	Very Low	Low	Nominal	High	Very High
Very Low	Very Low	Low	Moderate	High	Very High
Low		Very Low	Low	Moderate	High
Nominal			Very Low	Low	Moderate
High				Very Low	Low
Very High					Very Low

PMAT \ RELY	Very Low	Low	Nominal	High	Very High
Very Low	Very Low	Low	Moderate	High	Very High
Low		Very Low	Low	Moderate	High
Nominal			Very Low	Low	Moderate
High				Very Low	Low
Very High					Very Low
Extra High					

STOR Risk Rules

ACAP \ STOR	Nominal	High	Very High	Extra High
Very Low	Low	Moderate	High	Very High
Low	Very Low	Low	Moderate	High
Nominal		Very Low	Low	Moderate
High			Very Low	Low
Very High				Very Low

PCAP \ STOR	Nominal	High	Very High	Extra High
Very Low	Low	Moderate	High	Very High
Low	Very Low	Low	Moderate	High
Nominal		Very Low	Low	Moderate
High			Very Low	Low
Very High				Very Low

CPLX Risk Rules

		CPLX					
		Very Low	Low	Nominal	High	Very High	Extra High
ACAP	Very Low		Very Low	Low	Moderate	High	Very High
	Low			Very Low	Low	Moderate	High
	Nominal				Very Low	Low	Moderate
	High					Very Low	Low
	Very High						Very Low

		CPLX					
		Very Low	Low	Nominal	High	Very High	Extra High
PCAP	Very Low		Very Low	Low	Moderate	High	Very High
	Low			Very Low	Low	Moderate	High
	Nominal				Very Low	Low	Moderate
	High					Very Low	Low
	Very High						Very Low

		CPLX					
		Very Low	Low	Nominal	High	Very High	Extra High
TOOL	Very Low		Very Low	Low	Moderate	High	Very High
	Low			Very Low	Low	Moderate	High
	Nominal				Very Low	Low	Moderate
	High					Very Low	Low
	Very High						Very Low

TIME Risk Rules

		TIME			
		Nominal	High	Very High	Extra High
PCAP	Very Low	Very High	High	Moderate	Low
	Low	High	Moderate	Low	Very Low
	Nominal	Moderate	Low	Very Low	
	High	Low	Very Low		
	Very High	Very Low			

		TIME			
		Nominal	High	Very High	Extra High
ACAP	Very Low	Very High	High	Moderate	Low
	Low	High	Moderate	Low	Very Low
	Nominal	Moderate	Low	Very Low	
	High	Low	Very Low		
	Very High	Very Low			

		TIME			
		Nominal	High	Very High	Extra High
TOOL	Very Low	Very High	High	Moderate	Low
	Low	High	Moderate	Low	Very Low
	Nominal	Moderate	Low	Very Low	
	High	Low	Very Low		
	Very High	Very Low			

Appendix D: Expert-COCOMO Risk Assessment – NASA Project Data.

Project ID	Size (KSLOC)	Actual Effort (staff-mo)	Risk Level	Project Risk	Schedule Risk	Product Risk	Platform Risk	Personnel Risk	Process Risk	Reuse Risk
1	25.90	117.60	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	24.60	117.60	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	7.70	31.20	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4	8.20	36.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	9.70	25.20	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
6	2.20	8.40	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	3.50	10.80	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
8	66.60	352.80	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
9	7.50	72.00	Low	1.30	3.60	2.00	3.50	0.00	0.00	0.00
10	20.00	72.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
11	6.00	24.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
12	100.00	360.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
13	11.30	36.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
14	100.00	215.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
15	20.00	48.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
16	100.00	360.00	Low	0.30	0.00	0.00	0.00	0.90	0.00	0.00
17	150.00	324.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
18	31.50	60.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
19	15.00	48.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
20	32.50	60.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
21	19.70	60.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
22	66.60	300.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
23	29.50	120.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
24	15.00	90.00	Low	0.30	0.00	1.40	0.00	0.00	0.00	0.00
25	38.00	210.00	Low	0.30	0.00	1.40	0.00	0.00	0.00	0.00

Appendix D: Expert-COCOMO Risk Assessment – NASA Project Data.

Project ID	Size (KSLOC)	Actual Effort (staff-mo)	Risk Level	Project Risk	Schedule Risk	Product Risk	Platform Risk	Personnel Risk	Process Risk	Reuse Risk
26	10.00	48.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
27	15.40	70.00	Low	1.90	5.90	1.80	3.20	0.90	0.00	0.00
28	48.50	239.00	Low	1.90	5.90	1.80	3.20	0.90	0.00	0.00
29	16.30	82.00	Low	1.90	5.90	1.80	3.20	0.90	0.00	0.00
30	12.80	62.00	Low	1.90	5.90	1.80	3.20	0.90	0.00	0.00
31	32.60	170.00	Low	1.90	5.90	1.80	3.20	0.90	0.00	0.00
32	35.50	192.00	Low	1.90	5.90	1.80	3.20	0.90	0.00	0.00
33	5.50	18.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
34	10.40	50.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
35	14.00	60.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
36	6.50	42.00	Low	0.30	0.00	1.40	0.00	0.00	0.00	0.00
37	13.00	60.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
38	90.00	444.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
39	8.00	42.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
40	16.00	114.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
41	177.90	1248.00	Low	1.90	5.90	1.80	3.20	0.90	0.00	0.00
42	302.00	2400.00	Low	0.70	0.00	1.70	0.00	0.00	3.10	0.00
43	282.10	1368.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
44	284.70	973.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
45	79.00	400.00	Low	0.30	0.00	1.40	0.00	0.00	0.00	0.00
46	423.00	2400.00	Low	0.70	0.00	0.00	0.00	0.00	5.60	0.00
47	190.00	420.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
48	47.50	252.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
49	21.00	107.00	Low	3.40	0.00	11.80	0.00	2.20	0.00	0.00
50	78.00	571.40	Low	0.90	0.00	1.30	0.00	0.00	5.40	0.00

Appendix D: Expert-COCOMO Risk Assessment – NASA Project Data.

Project ID	Size (KSLOC)	Actual Effort (staff-mo)	Risk Level	Project Risk	Schedule Risk	Product Risk	Platform Risk	Personnel Risk	Process Risk	Reuse Risk
51	11.40	98.80	Low	0.90	0.00	1.30	0.00	0.00	5.20	0.00
52	19.30	155.00	Low	0.90	0.00	1.30	0.00	0.00	5.20	0.00
53	101.00	750.00	Low	0.60	0.00	2.90	0.00	0.00	0.00	0.00
54	219.00	2120.00	Low	0.70	0.00	3.00	0.00	0.00	0.00	0.00
55	50.00	370.00	Low	0.60	0.00	2.90	0.00	0.00	0.00	0.00
56	227.00	1181.00	Low	2.30	6.30	5.10	3.20	0.00	0.00	0.00
57	70.00	278.00	Low	1.00	2.70	1.40	0.00	0.90	0.00	0.00
58	0.90	8.40	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
59	980.00	4560.00	Low	0.30	0.00	1.50	0.00	0.00	0.00	0.00
60	350.00	720.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
61	70.00	458.00	Low	1.20	0.00	3.50	0.00	1.30	0.00	0.00
62	271.00	2460.00	Low	1.20	0.00	3.50	0.00	1.30	0.00	0.00
63	90.00	162.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
64	40.00	150.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
65	137.00	636.00	Low	0.30	0.00	1.40	0.00	0.00	0.00	0.00
66	150.00	882.00	Low	1.70	0.00	3.10	0.00	0.00	8.70	0.00
67	339.00	444.00	Low	0.70	0.00	3.20	0.00	0.00	0.00	0.00
68	240.00	192.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
69	144.00	576.00	Low	2.00	2.70	2.90	0.00	0.00	8.30	0.00
70	151.00	432.00	Low	1.60	2.80	1.30	0.00	0.00	8.30	0.00
71	34.00	72.00	Low	1.60	2.70	1.30	0.00	0.00	8.10	0.00
72	98.00	300.00	Low	1.60	2.70	1.30	0.00	0.00	8.30	0.00
73	85.00	300.00	Low	1.60	2.70	1.30	0.00	0.00	8.20	0.00
74	20.00	240.00	Low	1.60	2.60	1.30	0.00	0.00	8.00	0.00
75	111.00	600.00	Low	1.60	2.70	1.30	0.00	0.00	8.30	0.00

Appendix D: Expert-COCOMO Risk Assessment – NASA Project Data.

Project ID	Size (KSLOC)	Actual Effort (person-mo)	Risk Level	Project Risk	Schedule Risk	Product Risk	Platform Risk	Personnel Risk	Process Risk	Reuse Risk
76	162.00	756.00	Low	2.00	2.80	3.00	0.00	0.00	8.30	0.00
77	352.00	1200.00	Low	3.10	2.80	6.40	0.00	0.00	11.80	0.00
78	165.00	97.00	Low	3.10	2.80	6.30	0.00	0.00	11.60	0.00
79	60.00	409.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
80	100.00	703.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
81	32.00	1350.00	Low	0.30	0.00	1.40	0.00	0.00	0.00	0.00
82	53.00	480.00	Low	0.00	0.00	0.00	0.00	0.00	0.00	0.00
83	41.00	599.00	Low	0.70	2.70	0.00	0.00	0.90	0.00	0.00
84	24.00	430.00	Low	0.70	2.70	0.00	0.00	0.90	0.00	0.00
85	165.00	4178.20	Low	0.30	0.00	1.50	0.00	0.00	0.00	0.00
86	65.00	1772.50	Low	0.30	0.00	1.50	0.00	0.00	0.00	0.00
87	70.00	1645.90	Low	0.30	0.00	1.50	0.00	0.00	0.00	0.00
88	50.00	1924.50	Low	1.30	0.00	3.60	0.00	0.00	3.90	0.00
89	7.25	648.00	Moderate	7.00	0.00	11.70	7.20	8.50	2.70	0.00
90	233.00	8211.00	Low	0.30	0.00	1.50	0.00	0.00	0.00	0.00
91	16.30	480.00	Low	0.30	0.00	1.40	0.00	0.00	0.00	0.00
92	6.20	12.00	Low	0.30	0.00	1.40	0.00	0.00	0.00	0.00
93	3.00	38.00	Low	0.30	0.00	1.30	0.00	0.00	0.00	0.00

Appendix E: Fuzzy-ExCOM Risk Assessment – NASA Project Data

Project ID.	Size (KSLOC)	Actual Effort (staff-mo)	Risk Category	Project Risk	Schedule Risk	Personnel Risk	Process Risk	Product Risk	Platform Risk	Reuse Risk
9	7.50	72.00	Low	3.77	4.78	8.04	6.46	3.75	3.59	1.54
10	20.00	72.00	Low	4.43	5.29	7.72	9.56	4.55	4.36	1.54
11	6.00	24.00	Low	4.18	4.82	7.57	9.19	4.13	3.96	1.54
12	100.00	360.00	Low	4.27	4.86	7.69	9.56	4.25	3.96	1.54
13	11.30	36.00	Low	4.60	5.20	9.10	9.27	4.15	3.96	2.67
15	20.00	48.00	Low	4.22	4.83	7.62	9.34	4.18	3.96	1.54
17	150.00	324.00	Low	4.35	4.86	7.98	9.62	4.27	4.23	1.54
18	31.50	60.00	Low	4.23	4.88	7.63	9.62	4.28	3.62	1.54
19	15.00	48.00	Low	4.07	4.31	7.61	9.30	4.16	3.43	1.54
20	32.50	60.00	Low	4.35	4.88	8.06	9.62	4.28	4.05	1.54
24	15.00	90.00	Low	4.57	5.24	8.77	9.56	4.41	4.16	1.95
25	38.00	210.00	Low	4.88	5.57	9.23	9.99	4.77	4.87	1.95
26	10.00	48.00	Low	4.65	5.35	8.98	9.50	4.02	4.87	1.95
27	15.40	70.00	Low	3.99	5.63	8.88	6.65	4.02	2.41	2.17
28	48.50	239.00	Low	4.00	5.63	8.90	6.68	4.04	2.41	2.17
29	16.30	82.00	Low	3.99	5.63	8.88	6.65	4.02	2.41	2.17
30	12.80	62.00	Low	3.99	5.62	8.87	6.64	4.02	2.41	2.17
31	32.60	170.00	Low	4.00	5.63	8.90	6.67	4.03	2.42	2.17
32	35.50	192.00	Low	4.00	5.63	8.90	6.67	4.04	2.42	2.17
38	90.00	444.00	Low	4.36	5.32	9.32	7.30	4.23	4.80	1.54
40	16.00	114.00	Low	4.90	5.34	10.22	10.50	4.45	4.10	1.95

Appendix E: Fuzzy-ExCOM Risk Assessment – NASA Project Data

Project ID.	Size (KSLOC)	Actual Effort (staff-mo)	Risk Category	Project Risk	Schedule Risk	Personnel Risk	Process Risk	Product Risk	Platform Risk	Reuse Risk
1	25.90	117.60	Moderate	5.19	7.80	9.90	8.34	4.77	6.37	1.54
2	24.60	117.60	Moderate	5.19	7.80	9.90	8.33	4.77	6.37	1.54
3	7.70	31.20	Moderate	5.18	7.80	9.88	8.30	4.76	6.37	1.54
4	8.20	36.00	Moderate	5.18	7.80	9.88	8.31	4.76	6.37	1.54
5	9.70	25.20	Moderate	5.18	7.80	9.88	8.31	4.76	6.37	1.54
6	2.20	8.40	Moderate	5.17	7.79	9.85	8.27	4.74	6.37	1.54
7	3.50	10.80	Moderate	5.17	7.79	9.86	8.28	4.75	6.37	1.54
8	66.60	352.80	Moderate	5.20	7.80	9.92	8.36	4.78	6.37	1.54
14	100.00	215.00	Moderate	5.42	6.45	11.35	9.94	4.33	4.26	4.14
16	100.00	360.00	Moderate	5.27	5.67	11.43	10.02	4.44	4.28	3.50
21	19.70	60.00	Moderate	5.19	7.80	9.90	8.33	4.77	6.37	1.54
22	66.60	300.00	Moderate	5.20	7.80	9.92	8.36	4.78	6.37	1.54
23	29.50	120.00	Moderate	5.19	7.80	9.90	8.34	4.77	6.37	1.54
33	5.50	18.00	Moderate	5.17	7.79	9.86	8.30	4.74	6.37	1.54
34	10.40	50.00	Moderate	5.84	9.49	11.37	8.45	4.75	7.35	2.17
35	14.00	60.00	Moderate	5.18	7.80	9.88	8.32	4.76	6.37	1.54
36	6.50	42.00	Moderate	5.10	5.63	10.22	9.95	4.79	5.54	1.95
37	13.00	60.00	Moderate	5.02	5.57	10.16	9.83	4.44	5.54	1.95
39	8.00	42.00	Moderate	5.01	5.57	10.13	9.76	4.42	5.54	1.95
41	177.90	1248.00	Moderate	11.73	22.85	23.19	17.26	10.15	9.48	4.54

Appendix E: Fuzzy-ExCOM Risk Assessment – NASA Project Data

Project ID.	Size (KSLOC)	Actual Effort (staff-mo)	Risk Category	Project Risk	Schedule Risk	Personnel Risk	Process Risk	Product Risk	Platform Risk	Reuse Risk
42	302.00	2400.00	Moderate	12.06	14.66	25.49	17.30	12.65	15.51	4.36
43	282.10	1368.00	Moderate	9.12	13.81	18.43	12.36	6.31	12.74	4.36
44	284.70	973.00	Moderate	10.23	14.76	18.79	16.02	9.01	13.37	4.36
45	79.00	400.00	Moderate	12.86	16.14	23.23	26.26	12.56	14.36	3.42
46	423.00	2400.00	Moderate	14.30	18.82	20.96	41.30	7.77	14.43	3.42
47	190.00	420.00	Moderate	8.66	12.83	15.81	14.44	6.29	11.80	3.42
48	47.50	252.00	Moderate	9.71	14.40	17.97	15.00	8.18	13.49	3.42
50	78.00	571.40	Moderate	13.54	16.14	19.38	41.29	11.87	8.88	3.42
51	11.40	98.80	Moderate	13.99	19.23	19.02	38.81	13.49	10.41	3.42
52	19.30	155.00	Moderate	14.41	20.42	19.11	39.46	12.52	12.56	3.42
53	101.00	750.00	Moderate	12.77	17.84	22.15	24.75	17.07	11.14	2.32
54	219.00	2120.00	Moderate	14.04	19.30	25.71	27.73	16.33	11.35	4.36
55	50.00	370.00	Moderate	13.68	18.98	25.26	26.27	15.47	11.73	4.36
56	227.00	1181.00	Moderate	10.42	18.95	16.84	16.75	10.98	10.81	3.42
57	70.00	278.00	Moderate	14.08	23.27	29.48	20.46	11.47	14.91	5.48
58	0.90	8.40	Moderate	14.01	23.31	28.78	19.90	12.86	16.27	3.42
59	980.00	4560.00	Moderate	11.10	17.76	18.11	14.63	20.41	9.82	2.10
60	350.00	720.00	Moderate	12.04	13.62	18.43	33.51	10.47	11.48	2.32
61	70.00	458.00	Moderate	13.10	17.68	25.65	17.68	23.80	10.62	2.32
62	271.00	2460.00	Moderate	13.14	17.71	25.69	17.86	23.85	10.62	2.32

Appendix E: Fuzzy-ExCOM Risk Assessment – NASA Project Data

Project ID.	Size (KSLOC)	Actual Effort (staff-mo)	Risk Category	Project Risk	Schedule Risk	Personnel Risk	Process Risk	Product Risk	Platform Risk	Reuse Risk
63	90.00	162.00	Moderate	12.09	16.12	18.52	29.29	8.39	14.43	3.42
64	40.00	150.00	Moderate	12.01	16.06	18.48	28.89	8.32	14.43	3.42
65	137.00	636.00	Moderate	11.66	17.15	17.30	25.53	13.13	10.44	3.42
66	150.00	882.00	Moderate	11.89	17.97	17.31	25.57	13.14	11.26	3.42
67	339.00	444.00	Moderate	13.36	17.54	20.17	28.89	19.23	10.44	3.42
68	240.00	192.00	Moderate	11.94	17.11	18.24	28.32	6.76	15.24	3.42
79	60.00	409.00	Moderate	9.76	14.04	19.74	13.20	13.20	10.31	2.32
80	100.00	703.00	Moderate	9.77	14.05	19.75	13.26	13.22	10.31	2.32
82	53.00	480.00	Moderate	12.37	19.05	19.13	23.67	13.08	15.07	2.32
83	41.00	599.00	Moderate	14.48	22.79	30.87	21.86	12.11	10.34	10.03
84	24.00	430.00	Moderate	14.34	22.65	30.75	21.28	11.90	10.34	10.03
85	165.00	4178.20	Moderate	14.27	18.13	22.68	28.15	20.33	14.85	2.32
86	65.00	1772.50	Moderate	14.32	18.17	22.71	28.38	20.43	14.85	2.32
87	70.00	1645.90	Moderate	14.43	18.25	22.76	28.86	20.61	14.85	2.32
90	233.00	8211.00	Moderate	14.26	18.12	22.67	28.09	20.31	14.85	2.32

Appendix E: Fuzzy-ExCOM Risk Assessment – NASA Project Data

Project ID.	Size (KSLOC)	Actual Effort (staff-mo)	Risk Category	Project Risk	Schedule Risk	Personnel Risk	Process Risk	Product Risk	Platform Risk	Reuse Risk
49	21.00	107.00	High	19.39	21.42	35.71	36.47	35.66	11.03	4.36
69	144.00	576.00	High	18.29	28.14	20.62	50.65	19.25	15.48	2.32
70	151.00	432.00	High	16.18	25.21	20.23	46.14	11.35	15.48	2.32
71	34.00	72.00	High	16.61	26.18	20.18	45.45	14.27	15.48	2.32
72	98.00	300.00	High	16.92	26.55	20.38	46.99	14.48	15.48	2.32
73	85.00	300.00	High	16.87	26.50	20.35	46.78	14.45	15.48	2.32
74	20.00	240.00	High	15.59	24.50	19.85	43.21	10.94	15.48	2.32
75	111.00	600.00	High	16.09	25.10	20.17	45.67	11.28	15.48	2.32
76	162.00	756.00	High	17.07	26.73	20.48	47.74	14.59	15.48	2.32
77	352.00	1200.00	High	18.90	27.40	21.10	54.03	20.67	15.48	2.32
78	165.00	97.00	High	18.63	27.12	20.94	52.74	20.40	15.48	2.32
81	32.00	1350.00	High	18.52	21.33	41.55	29.18	20.15	19.23	6.71
88	50.00	1924.50	High	16.27	18.69	26.85	30.99	27.68	14.85	2.32
89	7.25	648.00	High	23.74	25.93	54.30	38.33	29.78	20.23	8.52
91	16.30	480.00	High	17.24	21.50	36.81	28.75	20.04	14.81	6.71
92	6.20	12.00	High	17.12	21.41	36.67	28.21	19.91	14.81	6.71
93	3.00	38.00	High	17.03	21.34	36.57	27.82	19.81	14.81	6.71

Appendix F: Effort Contingency Value – NASA'93 Project Data

Proj. ID.	Size (KSLOC)	Size Category	Project Risk	Risk Category	Actual Effort (staff-mo)	COCOMO II	RE	MRE	Contingency Allowance	Contingency Category	MIN Effort Est Value	MAX Effort Est Value
1	25.90	Small	5.19	Moderate	117.60	104.97	11%	11%	25.7%	Medium	77.99	131.95
2	24.60	Small	5.19	Moderate	117.60	99.49	15%	15%	25.7%	Medium	73.92	125.06
3	7.70	Small	5.18	Moderate	31.20	29.69	5%	5%	25.7%	Medium	22.06	37.32
4	8.20	Small	5.18	Moderate	36.00	31.70	12%	12%	25.7%	Medium	23.55	39.85
5	9.70	Small	5.18	Moderate	25.20	37.75	-50%	50%	25.7%	Medium	28.05	47.45
6	2.20	Small	5.17	Moderate	8.40	8.06	4%	4%	25.6%	Medium	6.00	10.12
7	3.50	Small	5.17	Moderate	10.80	13.06	-21%	21%	25.6%	Medium	9.72	16.40
8	66.60	Medium	5.20	Moderate	352.80	280.63	20%	20%	25.8%	Medium	208.23	353.03
9	7.50	Small	3.77	Low	72.00	24.82	66%	66%	25.0%	Low	18.62	31.03
10	20.00	Small	4.43	Low	72.00	36.80	49%	49%	25.0%	Low	27.60	46.00
11	6.00	Small	4.18	Low	24.00	11.04	54%	54%	25.0%	Low	8.28	13.80
12	100.00	Medium	4.27	Low	360.00	201.60	44%	44%	25.0%	Low	151.20	252.00
13	11.30	Small	4.60	Low	36.00	28.36	21%	21%	25.0%	Low	21.27	35.45
14	100.00	Medium	5.42	Moderate	215.00	479.18	-123%	123%	26.5%	Medium	352.20	606.16
15	20.00	Small	4.22	Low	48.00	39.40	18%	18%	25.0%	Low	29.55	49.25
16	100.00	Medium	5.27	Moderate	360.00	411.53	-14%	14%	26.0%	Medium	304.53	518.53
17	150.00	Large	4.35	Low	324.00	451.79	-39%	39%	30.1%	Medium	315.80	587.78
18	31.50	Small	4.23	Low	60.00	73.72	-23%	23%	25.0%	Low	55.29	92.15
19	15.00	Small	4.07	Low	48.00	29.07	39%	39%	25.0%	Low	21.80	36.34
20	32.50	Small	4.35	Low	60.00	122.25	-104%	104%	25.0%	Low	91.69	152.81
21	19.70	Small	5.19	Moderate	60.00	78.95	-32%	32%	25.7%	Medium	58.66	99.24
22	66.60	Medium	5.20	Moderate	300.00	280.63	6%	6%	26.0%	Medium	207.67	353.59
23	29.50	Small	5.19	Moderate	120.00	120.20	0%	0%	25.7%	Medium	89.31	151.09
24	15.00	Small	4.57	Low	90.00	57.99	36%	36%	25.0%	Low	43.49	72.49
25	38.00	Small	4.88	Low	210.00	163.26	22%	22%	25.0%	Low	122.45	204.08
26	10.00	Small	4.65	Low	48.00	30.94	36%	36%	25.0%	Low	23.21	38.68
27	15.40	Small	3.99	Low	70.00	66.08	6%	6%	25.0%	Low	49.56	82.60
28	48.50	Small	4.00	Low	239.00	218.17	9%	9%	25.0%	Low	163.63	272.71
29	16.30	Small	3.99	Low	82.00	70.10	15%	15%	25.0%	Low	52.58	87.63
30	12.80	Small	3.99	Low	62.00	54.50	12%	12%	25.0%	Low	40.88	68.13
31	32.60	Small	4.00	Low	170.00	144.27	15%	15%	25.0%	Low	108.20	180.34
32	35.50	Small	4.00	Low	192.00	157.65	18%	18%	25.0%	Low	118.24	197.06
33	5.50	Small	5.17	Moderate	18.00	20.91	-16%	16%	25.6%	Medium	15.56	26.26
34	10.40	Small	5.84	Moderate	50.00	40.60	19%	19%	27.9%	Medium	29.27	51.93
35	14.00	Small	5.18	Moderate	60.00	55.32	8%	8%	25.7%	Medium	41.10	69.54
36	6.50	Small	5.10	Moderate	42.00	31.54	25%	25%	25.4%	Medium	23.53	39.55
37	13.00	Small	5.02	Moderate	60.00	59.66	1%	1%	25.1%	Medium	44.69	74.63
38	90.00	Medium	4.36	Low	444.00	346.90	22%	22%	25.0%	Low	260.18	433.63
39	8.00	Small	5.01	Moderate	42.00	35.71	15%	15%	25.0%	Low	26.78	44.64
40	16.00	Small	4.90	Low	114.00	82.47	28%	28%	25.0%	Low	61.85	103.09
41	177.90	Large	11.73	Moderate	1248.00	1035.91	17%	17%	61.6%	High	397.79	1674.03
42	302.00	Large	12.06	Moderate	2400.00	1120.94	53%	53%	75.0%	High	280.24	1961.65
43	282.10	Large	9.12	Moderate	1368.00	830.26	39%	39%	67.5%	High	269.83	1390.69
44	284.70	Large	10.23	Moderate	973.00	994.21	-2%	2%	75.0%	High	248.55	1739.87
45	79.00	Medium	12.86	Moderate	400.00	272.93	32%	32%	40.1%	Medium	163.49	382.37
46	423.00	Large	14.30	Moderate	2400.00	904.51	62%	62%	75.0%	High	226.13	1582.89
47	190.00	Large	8.66	Moderate	420.00	382.38	9%	9%	55.8%	High	169.01	595.75
48	47.50	Small	9.71	Moderate	252.00	157.89	37%	37%	37.5%	Medium	98.68	217.10
49	21.00	Small	19.39	High	107.00	152.63	-43%	43%	38.1%	Medium	94.48	210.78
50	78.00	Medium	13.54	Moderate	571.40	339.63	41%	41%	39.9%	Medium	204.12	475.14

Appendix F: Effort Contingency Value – NASA'93 Project Data

Proj. ID.	Size (KSLOC)	Size Category	Project Risk	Risk Category	Actual Effort (staff-mo)	COCOMO II	RE	MRE	Contingency Allowance	Contingency Category	MIN Effort Est Value	MAX Effort Est Value
51	11.40	Small	13.99	Moderate	98.80	43.19	56%	56%	28.7%	Medium	30.79	55.59
52	19.30	Small	14.41	Moderate	155.00	75.95	51%	51%	30.7%	Medium	52.63	99.27
53	101.00	Large	12.77	Moderate	750.00	392.41	48%	48%	50.1%	High	195.81	589.01
54	219.00	Large	14.04	Moderate	2120.00	1015.82	52%	52%	75.0%	High	253.96	1777.69
55	50.00	Small	13.68	Moderate	370.00	208.40	44%	44%	35.5%	Medium	134.42	282.38
56	227.00	Large	10.42	Moderate	1181.00	752.33	36%	36%	75.0%	High	188.08	1316.58
57	70.00	Medium	14.08	Moderate	278.00	340.43	-22%	22%	38.1%	Medium	210.73	470.13
58	0.90	Small	14.01	Moderate	8.40	3.18	62%	62%	25.3%	Medium	2.38	3.98
59	980.00	X Large	11.10	Moderate	4560.00	5048.36	-11%	11%	75.0%	High	1262.09	8834.63
60	350.00	Large	12.04	Moderate	720.00	1248.08	-73%	73%	75.0%	High	312.02	2184.14
61	70.00	Medium	13.10	Moderate	458.00	550.87	-20%	20%	38.1%	Medium	340.99	760.75
62	271.00	Large	13.14	Moderate	2460.00	2564.75	-4%	4%	75.0%	High	641.19	4488.31
63	90.00	Medium	12.09	Moderate	162.00	116.17	28%	28%	25.0%	Low	87.13	145.21
64	40.00	Small	12.01	Moderate	150.00	52.80	65%	65%	34.9%	Medium	34.37	71.23
65	137.00	Large	11.66	Moderate	636.00	729.48	-15%	15%	53.5%	High	339.21	1119.75
66	150.00	Large	11.89	Moderate	882.00	1007.39	-14%	14%	55.1%	High	452.32	1562.46
67	339.00	Large	13.36	Moderate	444.00	2312.83	-421%	421%	75.0%	High	578.21	4047.45
68	240.00	Large	11.94	Moderate	192.00	363.67	-89%	89%	75.0%	High	90.92	636.42
69	144.00	Large	18.29	High	576.00	500.66	13%	13%	54.3%	High	228.80	772.52
70	151.00	Large	16.18	High	432.00	368.39	15%	15%	55.3%	High	164.67	572.11
71	34.00	Small	16.61	High	72.00	87.12	-21%	21%	35.4%	Medium	56.28	117.96
72	98.00	Medium	16.92	High	300.00	301.24	0%	0%	50.5%	High	149.11	453.37
73	85.00	Medium	16.87	High	300.00	258.60	14%	14%	43.8%	Medium	145.33	371.87
74	20.00	Small	15.59	High	240.00	42.15	82%	82%	31.6%	Medium	28.83	55.47
75	111.00	Large	16.09	High	600.00	264.84	56%	56%	51.1%	High	129.51	400.17
76	162.00	Large	17.07	High	756.00	727.12	4%	4%	57.2%	High	311.21	1143.03
77	352.00	Large	18.90	High	1200.00	1704.70	-42%	42%	75.0%	High	426.18	2983.23
78	165.00	Large	18.63	High	97.00	663.53	-584%	584%	57.8%	High	280.01	1047.05
79	60.00	Medium	9.76	Moderate	409.00	323.70	21%	21%	37.5%	Medium	202.31	445.09
80	100.00	Medium	9.77	Moderate	703.00	578.40	18%	18%	47.8%	Medium	301.92	854.88
81	32.00	Small	18.52	High	1350.00	1262.18	7%	7%	38.0%	Medium	782.55	1741.81
82	53.00	Medium	12.37	Moderate	480.00	593.55	-24%	24%	36.2%	Medium	378.68	808.42
83	41.00	Small	14.48	Moderate	599.00	331.43	45%	45%	34.4%	Medium	217.42	445.44
84	24.00	Small	14.34	Moderate	430.00	180.35	58%	58%	31.7%	Medium	123.18	237.52
85	165.00	Large	14.27	Moderate	4178.20	3109.49	26%	26%	57.8%	High	1312.20	4906.78
86	65.00	Medium	14.32	Moderate	1772.50	1078.89	39%	39%	37.2%	Medium	677.54	1480.24
87	70.00	Medium	14.43	Moderate	1645.90	1173.67	29%	29%	38.1%	Medium	726.50	1620.84
88	50.00	Small	16.27	High	1924.50	1039.79	46%	46%	36.8%	Medium	657.15	1422.43
89	7.25	Small	23.74	High	648.00	109.16	83%	83%	47.1%	Medium	57.75	160.57
90	233.00	Large	14.26	Moderate	8211.00	4602.45	44%	44%	75.0%	High	1150.61	8054.29
91	16.30	Small	17.24	High	480.00	287.53	40%	40%	33.3%	Medium	191.78	383.28
92	6.20	Small	17.12	High	12.00	94.43	-687%	687%	30.6%	Medium	65.53	123.33
93	3.00	Small	17.03	High	38.00	40.92	-8%	8%	28.8%	Medium	29.14	52.70

Curriculum Vitae

Name:	Ekananta Manalif
Post-secondary Education and Degrees:	Western University London, Ontario, Canada 2010-2013, MESc Student in Software Engineering University of Indonesia Jakarta, Indonesia 2000-2002, MSc in Computer Science (IS) Sepuluh November Institute of Technology Surabaya, Indonesia 1985-1990, B.Eng in Control System Engineering
Related Work Experience	Teaching Assistant Electrical and Computer Engineering Department Western University, Ontario, Canada 2010-2012 Lecturer School of Information System Binus University, Jakarta, Indonesia 2002-2007

Publications:

[1]. Manalif E., Capretz L. F., Ho D., "Software Project Risk Assessment and Effort Contingency Model based on COCOMO Cost Factors", *Journal of Computation & Modeling*, International Scientific Press Ltd., *accepted Feb-2013*.

[2]. Manalif E., Capretz L. F., Ho D., "Fuzzy rule-based for Risk Assessment and Effort Contingency Estimation within COCOMO Software Project Planning Model", Chapter in book: *Exploring Innovative and Successful Applications of Soft Computing*, *accepted Jul-2012*.

[3]. Manalif E., Capretz L. F., Ho D., Nassif A. B., "Fuzzy-ExCOM Software Project Risk Assessment", 11th IEEE Conference of Machine Learning and Application (ICMLA), Boca Raton, Florida, USA, December 2012, DOI 10.1109/ICMLA.2012.193.

[4]. Manalif E., Capretz L. F., Ho D., "Development of Integrated Software Project Planning Model Based on Cost Factors and Fuzzy Technique", *International Journal of Management Science and Information Technology*, NAISIT (North America Institute of Science and Information Technology), Issue 5, Jul-Sep 2012.